THE BUSINESS ONE IRWIN DESKTOP PUBLISHING LIBRARY

The Complete Scanner Toolkit Macintosh® Edition

DISK INCLUDED

DAVID D. BUSCH

The Complete Scanner Toolkit, Macintosh® Edition

David D. Busch

BUSINESS ONE IRWIN
HOMEWOOD, IL 60430

Sponsoring editor: Susan Stevens, Ph.D.
Project editor: Jane Lightell
Production manager: Mary Jo Parke
Printer: R. R. Donnelley & Sons Company

Library of Congress Cataloging-in-Publication Data

Busch, David D.
The complete scanner toolkit / David D. Busch.—Macintosh ed.
p. cm.
ISBN 1-55623-480-5
1. Optical scanners. 2. Printers (Data processing systems)
3. Desktop publishing. I. Title.
TK7882.S3B86 1992
006.6'2—dc20 91-13535

Printed in the United States of America

1 2 3 4 5 6 7 8 9 0 DOC 8 7 6 5 4 3 2 1

Dedication:

For Cathy, who's had nine books dedicated to her so far, at least that she'll tell me about.

About The Author:

David D. Busch has been writing about scanners since 1974 and using them in his own desktop publishing efforts since 1985. He has simplified confusing computer technology for millions of readers through best sellers like *Secrets of MacWrite, MacPaint, and MacDraw, The Complete Scanner Handbook for Desktop Publishing, Macintosh Edition*, and *Using DisplayWrite 4*. Two of his 32 books have earned top honors in the Computer Press Awards.

A former contributing editor and monthly columnist for six different computer magazines, Busch has written more than 2000 articles for publications as diverse as *Adam, Petersen's PhotoGraphic, Income Opportunities*, and *Writer's Yearbook.*

Vice-president of East Central Ohio Mensa, Busch and his wife have twice won the group's Es-Car-Go Road Rallye. The 1970 Kent State University graduate and spouse reside in their native Ohio with their four children, aged 2 to 22.

Acknowledgments:

This book wouldn't have been possible without the assistance, intentional or otherwise, of a number of people. These include:

Susan Glinert Stevens, Ph.D. at BUSINESS ONE IRWIN, who believed that the industry was ready for not one, but two new books dedicated to scanners for Macintosh users.

My college-age sons David Jr., and Michael, for making this book a financial necessity.

My two preschoolers, Jonathan and Teryn, for giving me something extra to look forward to in my twilight years.

Cathy, my wife and mother of all four, for making the 18 years between batches worth the wait.

And finally, thanks to the writers whose skill at storytelling has kept me, through a combination of awe and envy, from straying from the bounds of the nonfiction field: Norman Spinrad, Joe Haldeman, Mike Resnick, Piers Anthony, Cordwainer Smith, Tom Clancy, Nelson DeMille and, most of all, Robert A. Heinlein.

Contents

Preface

If you're a scanner user who is hungry for new ideas and clever techniques, you will find in this book the tools and tips you're looking for. It includes advanced information on such topics as color separating and halftones. You'll learn what to do before you scan to optimize an image and how to improve less-than-perfect scans once they have been captured. You'll find special tricks for using the latest image manipulation tools.

Best of all, this book includes a disk with some special applications. There's a demo version of Studio/1 Version 1.0, a sophisticated animation and painting program that will work with any Macintosh. QuickGIF and PCXtc are utilities that let you view and convert scanned images in a variety of formats. GrayView is a gray scale editing program with some unique features.

Introduction

Okay, I was wrong. In my last book about scanners, *The Complete Scanner Handbook for Desktop Publishing, Macintosh Edition*, I said there were certain things you couldn't do with a low cost scanner. Hundreds of thousands of you paid no attention to me and are doing those things anyway. You're making color separations and halftones with desktop scanners that cost less than $1900. You're capturing text with OCR programs and hand-held scanners. You're doing a lot of things that I said were technically feasible but not really a good idea given the limitations of the technology.

Since I wrote the first book, the horizons of scanning technology for the Macintosh have broadened considerably, and applications for scanners have increased significantly. Even though *The Complete Scanner Handbook* was the most advanced scanner guide on the market at the time, there was obviously room for another book—one that took a slightly different approach. You're holding that book in your hands.

The Complete Scanner Toolkit, Macintosh Edition offers advice that will help you get the most from your computer memory and other hardware, and it tells you things the vendor won't about installing your scanner.

More Than You Need to Know?

Previous scanner books from other publishers have been skimpy—usually no more than about 200 pages—and fairly general in approach. Several have tried to treat the Macintosh and IBM worlds in a single volume. If those authors were able to cover the scanner universe in so few pages, you may wonder why it has taken me two books (actually, four, since I've covered the very different Macintosh and IBM worlds in volumes of their own.) and nearly 1000 pages to discuss most of what I think you should know about scanners.

It's simple. While scanners are easy to learn to use, the more you know, the better you can use them. There are hundreds of software packages that can help scanner users work more efficiently, and dozens of hardware manufacturers have jumped on the scanner bandwagon with models of their own. Scanners automate complex tasks like color separating, halftoning, and data entry. The traditional methods often took years of study to master, and absorbing some of that lore can help you make better use of the computerized processor.

So, I divided the knowledge you need into two books for each platform. The first volume, *The Complete Scanner Handbook*, includes capsule evaluations of many of the most important scanners on the market. It provides a description of every key software package for image editing, autotracing, and OCR. I plan to update that book annually to keep pace with improvements in hardware and software technology. In the first book, you'll find tidbits about the history of scanning and in-depth discussions of the advantages and disadvantages of hand-held, flatbed, overhead, and sheetfed scanners.

The Scanner Handbook offers a heavy dose of theory mixed in with a rich compendium of practical tips and hints. I recommend it as a starting point for those who want to become familiar with scanner technology and who may need some help in setting up and expanding their scanner/computer configurations.

The Complete Scanner Toolkit takes a different approach. In this book, I don't attempt to survey all the available hardware or software. Instead, I concentrate on using one or two different scanners. I explain how two or three packages in each category work and how to get the most from them.

Until now, nothing in print has provided such detailed suggestions for what to do with a scanner once you've gained a certain basic level of proficiency with the technology. The manuals supplied with new scanners do a good job of showing you how to configure and operate a particular unit, and there are several books that explain the technology and clarify such confusing topics as gray-scale imaging and color separating. As I noted, my first book on scanning addresses these issues, and provides the information you need to decide which type of scanner and software is best for your needs.

If you already own a scanner you are happy with, understand the underlying technology as much as you care to, and feel ready to move on to specific applications, this book is for you. It is filled with practical information you can use today and tools to make your job easier.

The Complete Scanner Toolkit, then, will not concentrate on the theory and technology of scanning. I'll summarize the information you need to know in a few chapters at the beginning of this book. If you're interested in learning more, *The Complete Scanner Handbook* would be a good companion to this volume. But, you don't need it to benefit from this one.

All you really need is the interest and desire to get the most from your scanner and the belief that these machines are among the most useful peripherals available for Macintosh computers. You are reading the book I wish had been available when I started using scanners more than five years ago.

What's in This Book

At the heart of *The Complete Scanner Toolkit* are some utilities and demonstration programs, which provide valuable tools to get you started working with scanned images. I'm providing two applications which will run on any Macintosh, from the Mac Plus on up. Two others were written especially for Macs with gray scale or color display capabilities. So, even if you use your Mac only for line art, or view gray scale images in dithered form, there's something for you on the enclosed disk. If you do have a color capable Mac, you'll find even more to like. The demos, shareware, and freeware include Studio/1, PCXtc, QuickGIF, and GrayView. I evaluated hundreds of programs before settling on these four, which I believe are among the best available in their respective categories. I'll tell you more about each of these in the next section.

Also in this book, you will find a broad range of useful tips that you can apply to your work immediately—whether you use your scanner for desktop publishing, presentations, OCR applications, or other work. Topics covered in *The Complete Scanner Toolkit, Macintosh Edition* include the following handy tips and techniques:

- How to scan slides with a flat-bed scanner.
- How to create a halftone that overcomes the limitations of 300 dpi printers.
- A simple photographic trick that can double the contrast range of a scanned image.
- Compiling your own clip art library.
- Getting the most from PostScript and scanned images.
- Foolproof techniques for rescanning images that already have been made into halftones.
- Optimizing the resolution of scanned line art.
- Color separating for popular desktop publishing programs.
- Using scanned images in word processing documents.

Recent estimates call for several million potential scanner users by the end of 1991. By 1993, there should be 800,000 users of hand scanners, alone. Of these, nearly 600,000 are desktop publishers—one of the fastest growing segments of the computer industry. While sales of CPUs have leveled off, advanced peripherals like scanners and laser printers continue to generate excitement with their improved capabilities. In a recent magazine poll, scanners were voted one of the top three peripherals—much to the surprise of everyone but the vendors who market scanners, and the hundreds of thousands of new users of these handy devices.

Profit Without Honor

In my opinion, introductions to books like these should include at least a brief explanation of why the author feels qualified to ask the reader to spend good money for his or her opinions and knowledge. It's only fair to get this out in the open before you've wasted a lot of your time. There are very few people, other than myself, perhaps, who buy computer books purely for their value as escapist literature.

I'm probably the world's most successful unknown computer book author. No publisher has seen fit to grace the cover of a book with my picture, arms folded and sleeves rolled up. Nor does my name appear above or in the title, a la *DeVoney's Hard Disk Management*, in hopes that you'll purchase this book based on my reputation alone.

Yet, I have completed 30 or more of these books since 1983 and won two top awards from the Computer Press Association, including my very first book on the Macintosh, *Secrets of MacPaint, MacWrite, and MacDraw*, which I wrote in 1985.

I like to think that this success, coupled with a low profile, are attributable to my tendency to put myself in your shoes when writing these books. I'm less a celebrated industry pundit than a user, like yourself, in search of solutions. In the scanner arena, I'm fortunate enough to bring a unique image-oriented perspective to this never-ending quest.

My experience and training encompass both writing and professional photography. My first job that didn't involve dispensing Kool-Aid in paper cups to thirsty pedestrians was as a newspaper photographer. I'd applied for a job as a reporter/photographer while still in high school. The paper didn't need any reporters but wanted a part-time photographer who worked cheap. I responded by writing two- and three-page captions for my photos that soon obviated the need for a reporter to accompany me on assignments.

Later on, I served as a photo posing instructor with a Barbizon-affiliated modeling agency and operated a commercial studio/photo lab. I've published hundreds of articles on photography in the leading professional photo publications—from *Petersen's PhotoGraphic* to *The Professional Photographer*.

My interest in computers came somewhat later. The true pioneers of the industry consider me a newcomer, since I didn't get my first personal computer until 1977. But, I've purchased 20 or so since then and combined my interests and experience in books like this one.

About the Software

To make this a comprehensive toolkit, I'm including a disk with some valuable utility and applications software you can use to enhance your scanning activities. The disk includes the programs described below.

- Studio/1 is a sophisticated paint and animation program offered by Electronic Arts. The enclosed disk includes a special demo version with all the features of the program enabled—except the ability to print or save files. There's also an offer for readers of this book in Appendix D, which enables them to purchase the commercial version of this package—originally priced at $150—for only $55.
- PCXtc allows Mac users to view IBM PC-format black and white PCX files. For $12, you can upgrade to an advanced version that lets you convert these files, too.
- QuickGIF is a viewer for GIF files, such as the scanned images offered on CompuServe and other electronic BBS services. This is a shareware product. When you register for $30, the author provides an enhanced version for free.
- GrayView is the gray scale editor we all wish we'd had when we first got our Mac II's. The version included with this book has many advanced image manipulation features. If you don't have a gray scale editor, this program is an excellent place to start.

You'll find a longer and more detailed discussion of the shareware concept in Appendix B. In a nutshell, shareware is not free or public domain software; shareware is commercial, copyrighted software just like Microsoft Word, or even the Mac's System software. However, as anyone who has paid $695 for a box containing two disks, a thin manual, and 5 inches of foam padding can attest, packaging, marketing, and overhead can raise the price of useful packages beyond the reach of many who could benefit from them.

Shareware is not distributed through normal retail channels. Instead, these programs are made available through electronic bulletin board systems, user group libraries, and mail order shareware outlets, and often just between friends. You may elect to print your own manual from the file supplied with the program. Often, however, when you register the program, you receive a fancier bound manual.

In many cases, the type of distribution is the only difference between a shareware program and a program sold through retail channels. Many of the programs you see on store shelves were written entirely by a single person or a relatively small team of software engineers. Shareware, too, is often created by individuals or small companies.

And, just as there are some retail-distributed programs that are turkeys, you may find some really bad shareware. In either case, such programs tend to have a relatively short life. Word gets around that a given package performs poorly. Retailers won't carry bad software long, bulletin boards won't let bad shareware occupy valuable hard disk space, and unhappy users won't make copies to pass along to their friends.

Shareware Isn't Freeware

Another similarity between shareware and retail software is that you are expected to pay for it if you use it. With shareware, however, you can try the software first to ensure that it works with your computer and is compatible with other software. If you continue to use it after a trial period, you are required to register (purchase) the software. Since retail software packages that have been opened are rarely returnable, your only option there is to try the software at the store or on the computer of a friend who happens to have it. Which approach would you say is more convenient?

What shareware does is give new and interesting software a fair shot at success. You might be hesitant to pay $300 for software that does something you never knew you needed to do. You would, however, probably be willing to invest in a disk, some online time, or a book like this one for an opportunity to try a package that sounds interesting. Because the shareware vendor doesn't have huge amounts of money tied up in packaging, advertising, and distribution, he can give you that opportunity.

Since shareware often has the same features and functionality as conventionally distributed software, sometimes it ends up in retail channels. A package recognized as superior can reach many more potential users when sold at retail, because everyone just doesn't have access to a BBS, a user group, friends who collect software, or another shareware outlet. You pay for shareware by registering your package with the author or publisher.

Chapter Outline

1 The Scanner in Your Future

This chapter covers basic scanner theory in a few pages, then describes some of the applications for scanning. Desktop publishing, word processing, image data bases, presentations, and other applications are described.

2 How a Scanner Works

This chapter includes a quick rundown of how the various types of scanners work their magic. You'll learn how flatbed, sheetfed, overhead, and video scanners operate.

3 Scanners and Memory

How much memory you have may be more critical than you think. This chapter explains the different types of memory available for the Mac, under both System 6.0x and System 7, how scanners use them, and how to get the most from the memory you have.

4 Scanners and Disk Storage

This chapter examines some of the things that make disk storage so important to scanner users and tells how you can select the best hard disk for your needs.

5 The Hand Scanner Revolution

The hand scanner has become more than a low cost, entry level replacement for a desktop scanner. Today, most models can do nearly anything most users ever want!

6 What Vendors Don't Tell You About Installing Scanners

You follow the instructions—yet your scanner doesn't work. What went wrong? This chapter outlines what you can do at 10:00 p.m. on a Sunday night when the vendor's technical support department is closed.

7 Compiling Your Own Clip Art Library

You'll learn how to build your own collection of clip art from the tips in this chapter. Where to find clip art. How to scan it for best effect and how to modify it.

8 Manipulating Scanned Images

This chapter discusses some of the methods used in manipulating scanned images, including paint programs and raster-to-vector conversion software.

9 Converting Scanned Output

Why do you need to convert scanned images? This chapter describes some of the different file formats and tells why they exist, how to convert between them, and the benefits of multiple file formats.

10 OCR

This chapter is an introduction to the concept of optical character recognition. You'll learn about the different technologies available, and the strengths—and weaknesses—of each.

11 Getting the Best Halftones

This chapter explores some simple techniques for getting the best halftones. You are led step-by-step through the production of halftones from normal, high contrast, and low contrast original artwork. This chapter covers working with difficult images, including photos that have already been halftoned.

12 Introduction to Color Scanning

This chapter explains the confusing topic of color theory, and explains the most commonly used color models, such as RGB, CMYK, HSL, and Pantone systems.

13 Capturing Color with Your Scanner

How color scanners work, and how they can work for you. Within a few years, color scanners will be as common as gray scale scanners are today. You need to learn their capabilities now.

14 Color Separating with Your Scanner

This chapter discusses color separating in some detail. It covers color separating theory and then explains the advantages – and limitations – of color separating in desktop publishing.

15 Optimizing Output

This chapter covers how printers, typesetters and imagesetters handle your scanned output, and includes some tricks you can use to get better PostScript printing from scanned images – even if you don't have a PostScript printer. It also discusses using high resolution imagesetters for your output, when to use imagesetters.

16 Operating Environments and Scanners

Your choices involve more than just deciding between System 6.0x and System 7. Your system software can have a dramatic effect on your scanning activities. This chapter tells you why, and what you can do about it.

17 Other Tools and Techniques

This chapter covers some other tools, applications, and techniques for scanners that you'll find useful. Learn how to scan color slides with a flatbed scanner!

18 Using the Disk

Complete instructions for installing the programs on the disk included with this book.

19 Animation

This chapter will serve as your introduction to animation in general, and will offer a quick start for using Studio/1's impressive features.

20 Introduction to QuickGIF and PCXtc

Learn how to view and convert scanned images from a variety of formats with these two utilities.

21 Gray Scale Editing with GrayView

All the features of GrayView are discussed in this chapter. You'll learn how to manipulate TIFF, PICT, and ThunderScan files, and how to add gray tones to black-and-white MacPaint files!

Appendix A: Glossary

A comprehensive glossary with definitions of all the works you are likely to encounter whileusing your scanner.

Appendix B: What Is Shareware?

Tells you more about the shareware concept. Here, you'll learn why programs like QuickGIF can be made available for $30.00, when equivalent commercial programs may cost much more.

Appendix C: Resources

Addresses and phone numbers of some leading vendors of scanners, software, and other equipment.

Appendix D: Ordering Information

This appendix provides ordering information which you can use to upgrade and register the shareware included with this book.

1

The Scanner In Your Future

Predicting that there is a scanner in your future is about as risky as forecasting a laser in your living room or a hologram in your wallet. If you have a compact disc player or a Visa card, you have already made these high tech innovations a part of your daily life. If images are important to the work you do, a scanner is probably the next addition to your hardware inventory.

Why have scanners suddenly become so popular? There are dozens of reasons, the most important and obvious of which is that people want the services that only scanners can provide. However, we have wanted those services for years. Why are scanners only now exploding in popularity?

It's a truism of the computer industry that hardware always leads software capability by several years. In many cases, the time lag is considerably longer.

Scanner software is no different. While we now have microprocessors and System 7 software capable of rapid task-switching, and can expect true multitasking with System 8, scanner software used for capturing, editing, and outputting images is still oriented to a single task performed by a single user.

Yet, things are changing. Consider this list:

- Scanner software is starting to catch up to hardware, with more packages taking advantage of the high resolution full color graphics, more sophisticated memory management, and other capabilities of the latest generation of Macintosh equipment.
- More software is able to accept and manipulate scanned images. The Macintosh has always been far ahead of the IBM world in its use of

images, thanks to its built-in graphics capabilities and the Mac's Clipboard. Even very early applications like MacWrite could incorporate images created with MacPaint, and dithered images produced by early scanners, such as ThunderScan.

Yet, the more useful gray-scale and color images we scan and work with today couldn't really be used effectively until programs like ImageStudio, PhotoShop, and Digital Darkroom were introduced.

Today, leading WP programs, including Microsoft Word and Word Perfect, have incorporated features that make them mini desktop publishing programs. Even if you have nothing more than a word processor and a scanner, there are lots of useful things you can do.

However, you can now go far beyond those simple tasks. Desktop publishing programs from PageMaker and QuarkXpress to Ventura Publisher allow you to place scanned gray scale and line art and to specify a variety of halftone screen types and rules. Every drawing program, such as Adobe Illustrator and Canvas, offers an autotrace feature that converts scanned images to object-oriented art. There are even specialized gray scale and color image editing programs, such as the ones I mentioned earlier, along with ColorStudio, Studio/32, and PixelPaint Professional.

The scanned image is only a jumping off point. The exciting tools available today should increase the popularity of scanners significantly.

- The price of this advanced hardware has dropped sharply. A few years ago, a monitor and 24-bit color graphics card cost $3000 or more. Today, a usable high resolution monitor and card can cost as little as $1100 or even less.

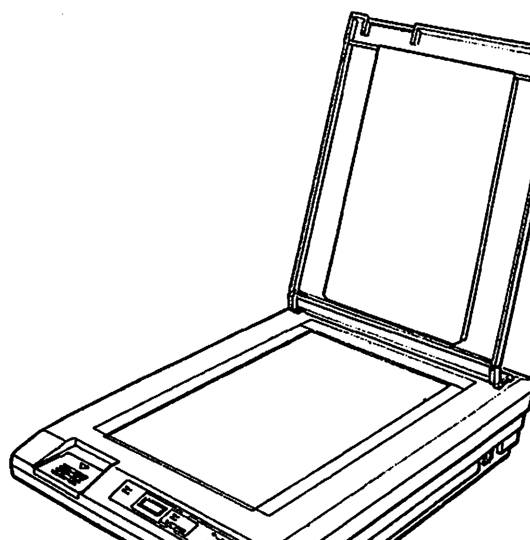
- Memory prices have fallen dramatically, too. Scanners capture and manipulate bit-mapped images; the higher the resolution, the more storage is required for the digitized image. For optimum speed, you want to be able to keep as much of an image as possible in memory, rather than on a hard disk, while you manipulate it. But even an image of modest size can occupy a great deal of memory.

Cheap memory means that today's scanning applications can be easier to use, and tomorrow's can boast new features and increased power.

- Scanners are getting better and cheaper. My first scanner, a Thunderscan model that attached to my Imagewriter, took up to half an hour to scan an 8 x 10-inch photograph with 16 levels of gray. Now you can scan a color photo with hundreds of thousands or even millions of individual colors in about 12 seconds. Resolutions of 300 to 400 dpi are standard, and 600 dpi units have started appearing on the market.

  And prices for even these powerful machines are dropping. My newest scanner, an Epson ES-300C color scanner, (shown in Figure 1.1)

Figure 1.1



The Epson ES-300C represents a new generation of scanners that provide full color capabilities at the same price as gray-scale-only scanners just a year or so ago.

lists for $1995 at this writing. Interfaces for the Macintosh and IBM PC list at $595 and $495 respectively.

Given the street price of this hardware, you could probably equip yourself with a color scanner like mine for less than $2000. I ended up with both Mac and IBM interfaces for it, because they can be used together to allow you to scan with both machines—alternatively, of course, not simultaneously. In effect, I got two color scanners for just a little more than the price of one.

You can also find several color scanners in overhead and other configurations for less than $1000. Hand-held scanners that capture 32 levels of gray cost only a few hundred dollars. Most recently, we've seen amazing color hand scanners like the Genius color scanner, which can capture a 4-inch swathe of 256 colors at 200 dots-per-inch resolution. A fancy mouse or keyboard can cost more than a basic scanner today.

- Scanners have become much easier to use. Hand-held models feature lights that warn you if you scan too fast. Scanning software can patch together strips of images so you can scan original artwork that is larger than your scanner's sensor width. It's now easier to preview images and zero in on the exact image you want to capture.

  Using scanners is no longer an arcane science, costly to implement and limited in capability. The scanner has become a universal tool for anyone who wants to add images to documents and presentations or capture text for other applications.

What Will It Cost Me?

If you already have a computer system you use for desktop publishing or word processing, adding a scanner can cost you very little. Every Macintosh system already has a graphics-capable monitor, a mouse, and enough memory to work with scanned images. All you need is the scanner itself, at a cost of $180 to $2000 for most types of work. Scanners that have extra features or are particularly adept at specialized tasks, such as color scanning, can cost more.

If you're starting from scratch and want to invest in a computer system that will meet your needs for scanning now and in the foreseeable future, set aside about $5000. That's Machrone's Law: The System You Want Costs $5000. That amount will get you a top-notch system with few compromises. I'll list the equipment required shortly. The dollar figure, though, was determined by applying what I'll call Busch's Corollary to Machrone's Law.

(Once writers gain a position of respect or authority, they often attempt to immortalize themselves with a law or rule of thumb. Jerry Pournelle, the science fiction writer and columnist for *Byte* and *InfoWorld* often cites Pournelle's Law: One user, one CPU. His SF colleague Theodore Sturgeon is remembered for Sturgeon's Law: Ninety Percent of Everything is Crap. Since each writer is allowed only one Law (except Isaac Asimov, who somehow got Three Laws of Robotics), I'm not going to waste mine on a refinement of an idea that industry writer and editor Bill Machrone and I developed independently, but which he rushed into print first.)

Busch's Corollary to Machrone's Law states: Anything Worth Doing Costs $5000. I formulated this axiom after assembling a first rate stereo system, collecting a set of Nikon cameras with a minimal complement of lenses, taking a trip to Europe, and putting together an efficient scanning system. Each of these cost roughly $5000. Obviously, vendors have figured out that $5000 is an optimal figure for disposal of discretionary funds. If the amount were pegged any higher, you would have to resort to a long term loan, with all the pain and undesirable forethought that requires. Any less than $5000, and the vendor isn't soaking up all the available funds.

To date, Busch's Corollary has just one advisory: The $5000 figure may be adjusted upward from time to time to account for inflation but is never adjusted downward to accommodate advances in technology. That is, if the $5000 decent scanner system of today can be purchased a year from now for $2500 it will no longer be considered acceptable. The new minimum will embrace many more capabilities and features than the old system and will still cost $5000.

Today, your $5000 will buy you a flatbed gray scale or color scanner, a Macintosh LC or IIsi computer with 4 Mb of memory. For the best results, you'll want to upgrade the LC's video memory to 512K or more to provide 8- to 16-bit color. Reviews in magazines like MacUser or MacWorld are your best guide to choosing a particular product. As I noted in the introduction, I also cover a variety of scanners in *The Complete Scanner Handbook.*

What Can I Do with a Scanner?

Many of the applications for scanners involve desktop publishing. However, scanners also have many uses in image and document management and word processing. This section highlights some of the uses for scanners that you may not have thought of.

- Capture text in a word processing environment. Even today, not all text is input by the executive, administrator, or clerical worker who creates it. In some organizations, many documents are dictated. Others are typed by an executive who has a typewriter at home but no computer. Still other documents may be received from outside the organization in hardcopy form. If you have a document created on an IBM PC and no easy way to exchange ASCII files between the PC and your Mac, scanning a hard copy can provide the solution.

  Scanners and optical character recognition (OCR) software can allow word processing personnel and others to edit and revise documents that are available only as paper copies.

- Capture text for desktop publishing. All the scenarios described above also apply to desktop publishing. A national group I belong to has several hundred local chapters throughout the United States. Most of these publish their own newsletters and exchange those publications with other groups. Half the content of many of them is made up of reprints from the other newsletters. OCR scanning provides a quick way to incorporate text from clippings, tear sheets, and other sources into a desktop publication.

I was once unlucky enough to lose a disk file of a publication of which I had only one proof copy available. Scanning in the text helped me avoid doing the job over from scratch.

- Manage documents. In some documents, valuable information isn't confined to the text. You may need to retain drawings, charts, and other images in the document. Perhaps a signature must be kept available for verification. A scanner in graphics mode can capture everything on a page—text, graphics, and formatting information; it's all there when you need it later. If you should decide later that the text itself is important, OCR programs can extract text from a page image that was stored as a bit map. Figure 1.2 shows a document that includes many different types of information.

- Fax more efficiently. A fax machine is not much more than a smart modem with a printer and a scanner attached. Since you already have a printer and a scanner, you can turn your computer into a fancy fax machine just by installing a fax modem—a modem that knows the special protocols used to exchange fax images.

  Why do you need a $3000 fax machine? Using your scanner and fax board lets you bypass long lines at the department fax machine. You can scan sensitive documents yourself and transmit them without exposing the information to unauthorized eyes. You may even get better quality. Some of the latest software on the market allows you to convert scanned images to PostScript instructions which can be faxed with less loss of resolution.

- Create your own fonts. You can scan images of characters you like and transform them into PostScript. Altsys and Letraset each have programs that let you do this. It's a lot of work to create an entire character set in this way, but it is an entirely practical way to develop a few letters to be used in a logo or a banner for your publication. Figure 1.3 shows an example of what you can do.

- Capture images to be merged into a desktop publication. These can be categorized in two groups: line art and continuous tone art like photographs.

The Myth of Mensa Intelligence

Ruling line

s frequently view of our e of the most widely-held misconceptions about Mensans is that they are smart. No one knows how the rumor got started. Some Mensans have even been known to succumb to the fallacy.

This column has always tried to deal with topics from a Mensa perspective. We figure that if all you want is a more erudite and pretentious version of columns that appear in run-of-the-mill publications, you'll subscribe to Vanity Fair or maybe Spy. So, this month, we'll provide you with the ammunition needed to dispel this vicious myth of Mensa intelli-

Body text

cant proof can eal-world ob- t:

1. Mensans almost never are seen purchasing wrapping paper and Christmas decorations at K-Mart on December 26. Instead, they buy these items at three times the price in the three weeks before Christmas, just like others of nominal intelligence. Being Jewish is no excuse, either. A bargain is a bargain.

2. "If you're so smart, why ain't you rich?" Few Mensans are really wealthy. One can only draw two possible conclusions from this: 1.) The average Mensan isn't intelligent enough to figure out that being rich is be

Headline

2.) Me enough t problem o money.

3. Regional gatherings.

If empirical evidence isn't sufficient, there has been a great deal of scientific thought devoted to the concept of intelligence. Researchers now believe that human intelligence actually can take one of seven forms, six of which are rarely, if ever, exhibited by Mensans. These types of "smarts" are:

1. Kinesthetic intelligence.

Believe it or not, Mike Tyson is actually a genius on th order of Leonardo Da Vinci, you allow that the functionin part of his brain is located somewhat lower on the spinal chord.

2. Testibular intelligence. This is the ability to do especially well on stupid tests, such e Stanford Binet or Miller ogies test. Unlike the forms of human intelligence, this particular skill has not been shown to have a useful purpose.

3. Crackheuristics. Well-known crackheurs are Southerners Ted Turner and Junior Johnson. This is a special form of business acumen well-suited for extracting sums of money from more intelligent Yankees.

4. Bon joviality. The ability to produce No. 1 record albums following brain death. First recognized in David Lee Roth.

5. Erudition. Affinity for cultural, sociological, political and quasi-scientific information of no practical value.

6. Problemathematical intelligence. Skill at balancing checkbooks and figuring tips. Rarely seen in Mensans, who have no need for checkbooks and don't leave tips.

7. Common sense. I'm not l about,

Line art

me it to in-

Figure 1.2

Scanners can capture everything on this page, and sophisticated software can separate a page into text and graphics, retaining formatting information such as columns, along with font size and attributes.

Line art consists of logos, charts, cartoons, and other artwork composed of lines. You can scan such art as a bit-mapped image, in which case the resolution will decrease as you enlarge the image in your publication. You can also trace the bit map using a drawing program or an autotrace facility to produce outline-oriented art that can be enlarged to any size without loss of resolution.

Continuous tone images can be in color or black and white and are translated into halftone images after scanning to allow reproduction on printing equipment. There is also a limited number of output devices with continuous tone capabilities that allow your scanned images to be reproduced in their full range of tones and colors.

- Capture images for display or positioning only. Even if you go the traditional route for producing color separations and halftones, your scanner can produce images that can be inserted in a dummy or mockup

Figure 1.3

Kitchen Table International

You can prepare your own fonts by scanning them in and modifying the characters with an image editing program.

of a publication or document to show what the finished piece will look like. This "for position only" (FPO) artwork can speed the approval process and eliminate the need for some people to review the stripped in, typeset copy before the printing plate is made.

- Provide a low cost, lower resolution substitute for an original image. Picture buyers can review a scanned image of a photograph, perhaps supplied on a CD-ROM, and then decide which images they want to purchase from a photo stockhouse. The ordered slides or prints can be used for the actual publication.

  Or, home buyers may want to review scanned images of the available properties before going out to visit a given house. The scanned image can provide enough information to eliminate some properties and move others to the top of the list. Electronic images of this type can be sorted and retrieved automatically using the home buyer's own criteria, so the process is much faster than poring over a book full of tiny photographs.

Summary

This chapter explains why I think there is a scanner in your future if you aren't using one already. While there has been a need for these devices for many years, their popularity has exploded in recent months.

There's more to the scanner explosion than the usual lag of software capabilities behind hardware developments. Certainly, scanner software is now starting to catch up to hardware, with more packages taking advantage of the capabilities of the latest generation equipment. More of this software can accept and manipulate scanned images. When scanners were first introduced, there wasn't a lot you could do with a scanned image. Now leading WP programs have incorporated features that make them mini desktop publishing programs. Even if you have only a word processor and a scanner, there are lots of useful things you can do.

The price of scanner hardware has dropped sharply, and memory prices have fallen dramatically as well. You can now buy full-color scanners for little more than a gray scale scanner cost a few years ago.

Cheap memory means that today's scanning applications can be easier to use, and tomorrow's can boast new features and increased power.

Scanners have also become much easier to use. They are a universal tool for anyone who wants to add images to documents and presentations or capture text for other applications.

There are many different applications for scanners. You can:

- Capture text in a word processing environment.
- Capture text for desktop publishing.
- Manage documents.
- Fax more efficiently.
- Create your own fonts.
- Capture line art or continuous tone images to be merged into a desktop publication.
- Capture images for display or positioning only.
- Provide a low cost, lower resolution substitute for an original image.

2

How A Scanner Works

You don't need to know how a scanner works to use one, particularly for the sort of work most people need to do. People who don't scan troublesome originals or who aren't faced with very high quality standards can safely ignore the technical details in this chapter. That's one of the reasons scanners have become so popular today. You don't have to be a technical guru to use one effectively most of the time.

However, you probably bought this book because you know that understanding what is going on can help you do a better job of scanning. If you know what vibration can do to the scanning head, you won't need to be told not to tap your fingers impatiently on the cover of your scanner as it works. Nor will you need the handholding that some users require when faced with a particularly difficult scanning job. You'll understand the limitations of your scanner, and you'll know how to overcome them. This chapter isn't technical on a mind-numbing level (I've saved that for the next chapter and the one that follows it). It just provides a brief introduction to how your scanner works.

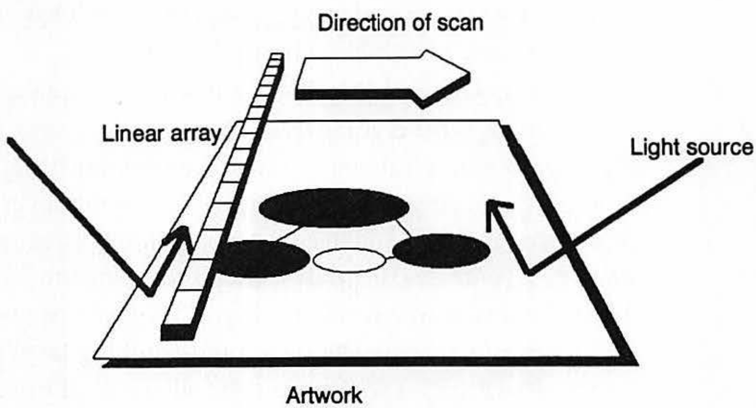
Types of Scanners

There are five basic types of scanners: flatbed, sheetfed, overhead, hand-held, and video. All five have several elements in common and some unique aspects of their own.

Among the things they share are a light source and a sensor that receives light reflected by or transmitted through the subject being scanned. The sensor is actually a sensor array — many small individual sensors arranged in a line or matrix. Usually, the sensor is a silicon chip called a charge-coupled device, or CCD. In the future, solid state scanners may be based on lower cost sensors using amorphous silicon technology. Figures 2.1 and 2.2 show the two types of sensors.

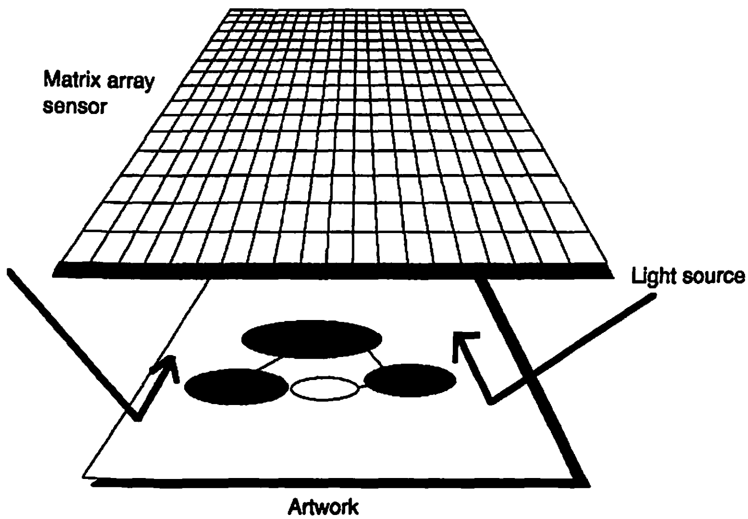
The more individual elements the CCD has, the higher its resolution is. At this writing, the highest resolution solid state sensor is a 4 megapixel array developed by Eastman Kodak Company. This expensive chip is capable of resolving 2048 x 2048 picture elements, or pixels.

Figure 2.1



A linear array of sensors is used in most scanners. Light is reflected from the artwork you are scanning up to the sensor bar. The array itself can be moved to cover the entire subject, line by line, or else you can keep the sensor stationary and move the artwork past it, as with a sheetfed scanner or fax machine.

Figure 2.2



A matrix array sensor is used in expensive scanning devices which can capture an entire image at once, as well as in video cameras. Such a scanner has no moving parts.

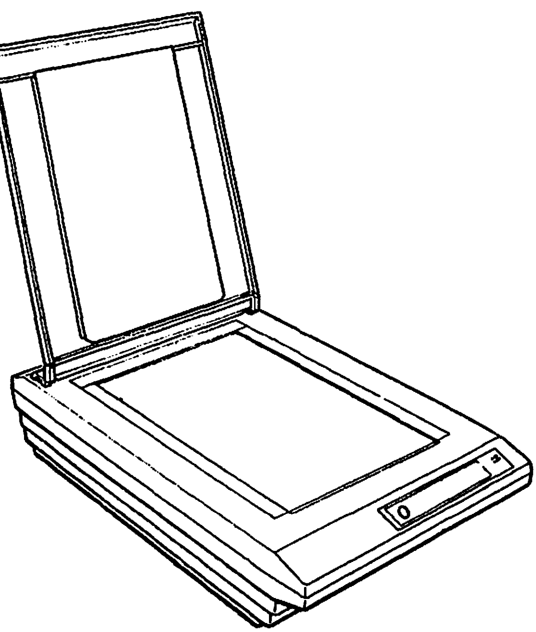
In technical terms, a CCD is a junctionless semiconductor. Like all semiconductors, it is a multilayer device coated on a silicon base, or substrate. One of these layers is silicon dioxide capped with closely spaced electrodes. Light reflected or transmitted from the scanned image is focused by a lens so it falls on an array of photodiodes, which output a voltage that varies according to the amount of light received. The current is applied to the electrodes in the CCD, generating a difference in electrical potential in the area beneath each electrode.

Each element of the array captures information about a single pixel in the image. That information is based on how much light is transmitted or reflected by the original. If very little light reaches the sensor, the pixel is stored as a dark tone. If a great deal of light makes its way to the sensor, the pixel is represented by a light tone.

A scanner that has a linear array scans one line of pixels at a time. Then, the next line is scanned, until all the lines in an image are captured. The sensor can change its point of view from one line to another in a variety of ways.

On a flatbed scanner, you place the original face down on a piece of glass, much as you would on a photocopier. Then, a traveling bar passes under the glass, scanning each line in turn. The bar has a light source that illuminates the original and either the sensor itself or a mirror that reflects light to a fixed sensor at the end of the scanner. Figure 2.3 shows a typical flatbed scanner.

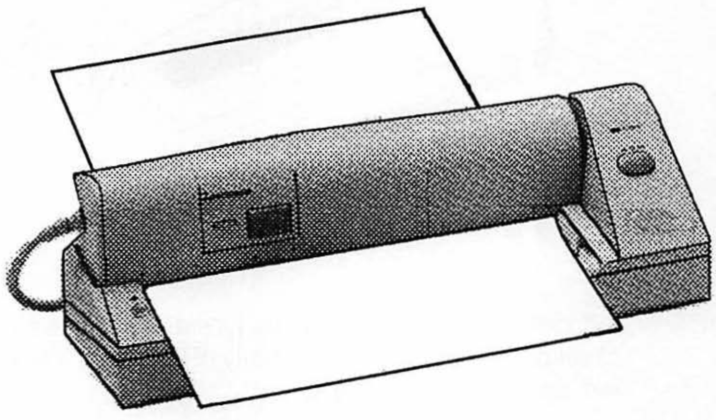
Figure 2.3



Flatbed scanners are available in both color and gray-scale versions, and have become ihe standard configuration for desktop publishing, OCR, and many other types of work.

With flatbed and sheetfed gray scale scanners, the light source is likely to be a full spectrum fluorescent light source. Color scanners use either three colored fluorescent lights or a system of filters to change the color of a single light source. In a sheetfed scanner, the sensor and the light source remain fixed, and the original image is pulled past them using a roller transport system. Figure 2.4 represents a sheetfed scanner.

Figure 2.4

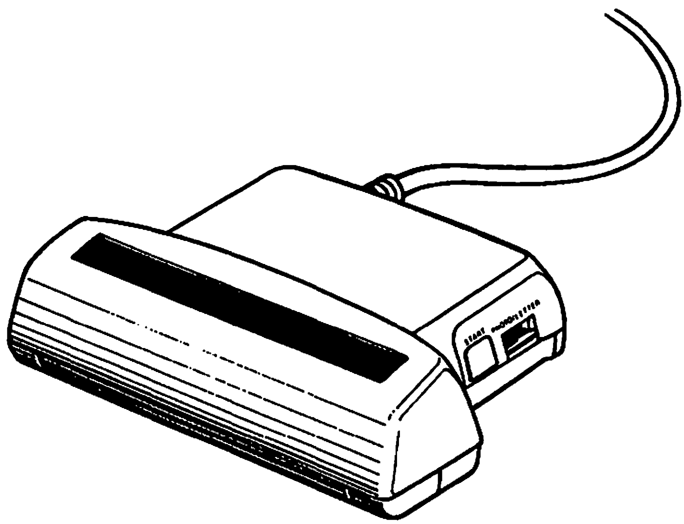


A sheetfed scanner, in which the sensor and light source remain fixed, while the document is fed through a system of rollers.

Hand-held scanners use your muscles to pull the scanning head past the material being scanned. In that respect, they resemble sheetfed models more than flatbed scanners, even though the artwork remains stationary. Like sheetfed scanners, hand-held models will produce bad scans if transport is not smooth and at right angles to the edge of the page. The fact that it is the scanner rather than the paper that is the variable makes no difference.

Hand-held scanners, like the one shown in Figure 2.5, lack both the space and the power needed for a fluorescent light source. Instead, a bank of light emitting diodes (LEDs) is used. Unfortunately, the LEDs

Figure 2.5



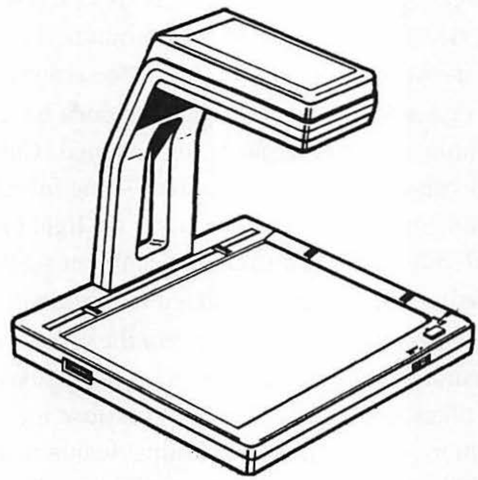
A handheld scanner. These are typically the lowest in cost of all types of scanners, yet they can do many of the same tasks as flatbed, sheet fed, and overhead models.

suitable for scanners don't emit a pure white light. The most commonly used LEDs provide red, green, or yellow illumination. Scanners which use red LEDs offer poor reproduction of reds and skin tones, so there has been a trend toward units that use green or yellow LEDs in recent months.

Overhead scanners place the sensor above the copyboard that holds the original, much like the arrangement found in a photographic copy camera. The sensor may be a linear array that is rotated to scan each line or that views a rotating mirror to achieve the same effect. Figure 2.6 shows an overhead scanner.

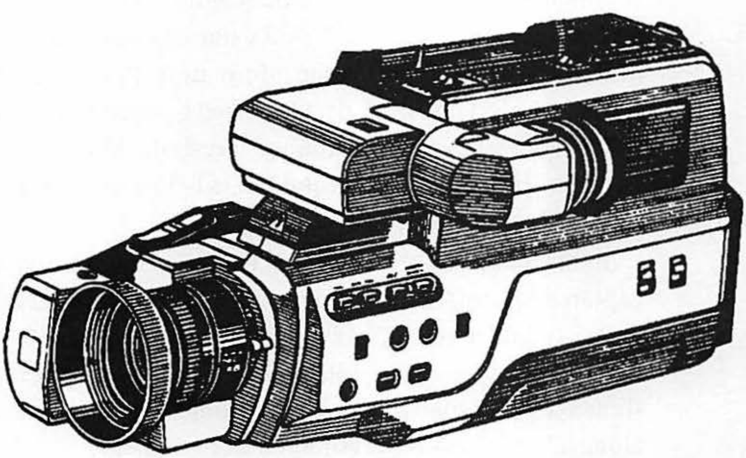
Some overhead scanners and all video scanners use a two-dimensional array of sensors to capture an entire original image at one time. This arrangement allows much faster scanning, of course, since the actual line-to-line scan portion of the process is done electronically. However, as you might expect, the sensors used in such scanners are

Figure 2.6



An overhead scanner

Figure 2.7



A video "scanner"—an amateur camcorder

either expensive or offer relatively low resolution because of the complexity of building a high resolution two-dimensional array. Figure 2.7 on the previous page shows a video scanner.

Flatbed and sheetfed scanners must have an internal light source to illuminate the material being scanned. Color models use either three individual colored light sources—one for each of the primary colors—or a system of filters placed over the light or sensor. Overhead scanners and video scanners often use ambient light but may work better if an auxiliary light source is added to illuminate the subject.

Higher light levels may allow the scanner to interpret dark tones more accurately. There is less room for confusion over whether a pixel should be black because it is black or because the scanner didn't have enough light to read the area in question. Scanners that must work with varying light levels adjust for changing light conditions automatically. Auxiliary lighting for overhead and video scanners also allows the illumination to be adjusted for three-dimensional objects and to avoid unwanted shadows on the subject.

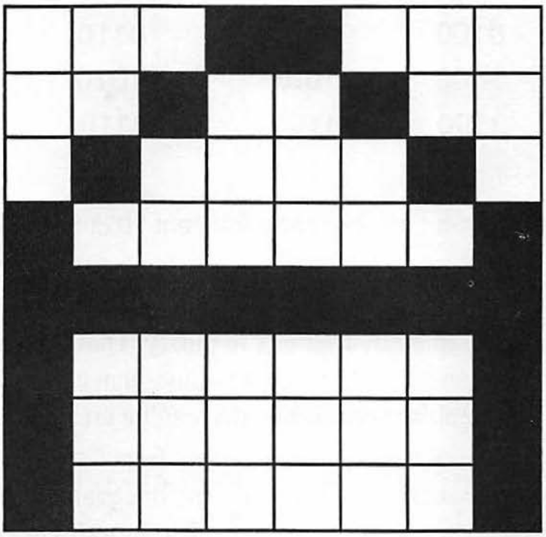
The information that a scanner captures is converted from a continuous, or analog, value to a digital value before it is stored. Scanners and their software can interpret the information they capture in several ways. At the simplest level, the data captured by each sensor can be measured to see if it passes a predetermined threshold. If so, the pixel is recorded as white. If the light level captured is below this threshold, the pixel is recorded as black.

In this mode, only a single binary digit, or bit, is required to store the captured information for a given pixel. A white pixel can be represented by a 1, while a black pixel can be represented by a 0. By storing each line separately, we can built a bit map of the original image. Figure 2.8 shows a very coarse, 8 x 8-pixel image of a scanned uppercase A, alongside a binary representation of that image.

In this example, the computer would collect each 8-bit line of information and store it as a single 8-bit byte in one memory location. Our binary image would require exactly 8 bytes of storage.

Because the scanner can move its threshold upward or downward, the same scanned analog information can be interpreted in different

Figure 2.8



A bit map of the letter A, and the binary numbers which represent the on/off pixels in the image.

ways. If the threshold is adjusted in one direction, some lighter toned pixels will be stored as black, rather than white. Moving the threshold in the other direction will classify pixels that had been interpreted as white to become black instead.

Because only a single bit is required to store any tone in the spectrum, this mode is usually called 1-bit, or binary, mode. As you can see, the scanner is still capable of sensing a broad range of grays or colors even though only black and white values are stored. If that were not so, scanners would be unable to handle light or dark toned originals in 1-bit mode.

The number of bits used to store scanned information is often referred to as pixel depth. You'll see references to 4-bit, 6-bit, 8-bit, and even 24-bit scans. All that means is that the analog information captured by the scanner is translated, or digitized, into the specified number of values. The more values available to represent a continuous tone image, the more accurate the final image.

Figure 2.9

0000	0001	0010	0011
0100	0101	0110	0111
1000	1001	1010	1011
1100	1101	1110	1111

A four-bit number can represent 16 different values

For example, in 4-bit mode, all the tones from black to white are represented by four bits in binary. That provides 16 different combinations and 16 different gray values that can be stored. Figure 2.9 lists the 16 combinations of bits that can be created with four binary digits:

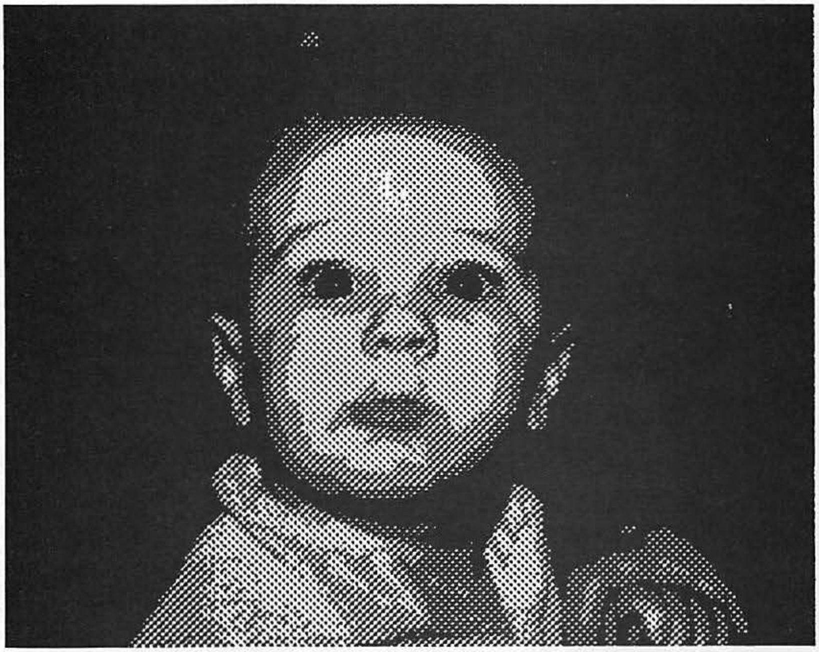
These represent the numbers from 0 to 15 in decimal notation. The 16 tones can be evenly distributed, or equalized, or clumped together at one end of the gray scale or the other. In addition, the threshold that separates black from some other tone can still be moved up and down as you want. Moving the threshold up has the effect of brightening the image, if the distance between values remains the same. Even with only 16 tones, you can change the contrast and brightness of a gray scale image.

Early scanners could capture only 4 bits per pixel. However, we soon had scanners that divided scanned data into 64 values and required 6 bits to store that data. I won't list all the combinations; you'll have to take my word for it.

Today, the standard is a pixel depth of 8 bits, which provides 00000000 to 11111111 binary—0 to 255—or 256 different gray levels. Since the human eye can differentiate only 30 to 60 different gray levels at one time, that's more than we need to represent vivid, lifelike black-and-white images. If you use fewer tones to represent a continuous tone image, your software will group similar tones together as one of the available values.

If these tones in your original subject are scattered evenly throughout the image, even 16 different tones may be quite enough to show the image realistically. For example, some scenes and many human por-

Figure 2.10



A posterized image, in which all the gray tones have been compressed into a limited number of grays.

traits are filled with details that break up the image into thousands of tiny areas that vary in tonal value.

If an image has large areas of tone that change gradually from one to another (a so-called gradient), you'll need many more shades of gray to provide a smooth transition. Otherwise, there will be abrupt changes between tones, which produce the banding effect that is called posterization. At times, this effect can provide an interesting representation of an image that looks something like a circus poster. Most of the time, though, we want to avoid it. Figure 2.10 shows a posterized image.

It's possible to scan a color image three times and store 256 tonal values for each of the three primary colors of light. Those 24 bits of information (three times 8 bits) can be used to produce more than 16.7 million different colors. Only the most expensive video cards and monitors available for the Macintosh are able to show that many colors at once. I use one of the lower cost solutions: a RasterOps ColorBoard 264 (about $500) and a Sony 1304 MultiSync HG monitor ($650) to view 24-bit color on my Mac II. Screen redrawing is a little slow, but I can afford this equipment. In 256-color mode, 24- and 32-bit color is generally converted to a palette of, say, 256 colors, for display. We'll look at color scanning in more detail later in this book.

It's important to note that the pixel depths described so far are more theoretical than practical. Some information is always lost due to a variety of factors, including the need to differentiate between valid data (the signal) and the background garbage produced by the system itself (the noise). If you're an audiophile or an engineer, the concept of signal-to-noise ratio should be familiar to you.

An 8-bit scanner that should be able to capture 256 gray tones, may actually be able to sense only 64 gray tones and should rightly be classified as a 6-bit scanner. Dark areas will go black, and light areas will go white. This is why we are now seeing on the market scanners that capture 12 bits of data; they actually produce 8 bits of useful information.

Scanners and Resolution

Resolution is properly defined as the amount of information per linear inch or square inch that can be captured or reproduced by a system. Too often, though, we think of resolution solely in terms of the number of dots per inch, which can be misleading. Most printers can produce only a single size and density dot, so the 300 dpi ratings you see attached to laser output devices really tell you that these printers can pack up to 300 fixed size dots — or white spaces — in a given linear inch, or up to 90,000 of them in a square inch.

Many of the images we scan are not strictly black-and-white. Continuous tone images represent dozens or thousands of different gray tones. The ability of a scanner to capture and represent these tones is an important part of its resolution. An image scanned at 300 dpi in 1-bit, binary mode will appear crude compared to one scanned at 300 dpi in 4-bit mode. Instead of 90,000 dots in a square inch, a 4-bit scan produces

1.4 million possible dots — 16 times as much information. Your printer may not be able to reproduce that square inch faithfully, but the screen of your 8-bit or 24-bit monitor can display it quite well.

Because the gray scale component of an image is an important part of its resolution, a 200 dpi continuous tone printer will produce images that are much more lifelike and full of detail than a 300 dpi binary output device like your laser printer. We'll look at how grays and colors are simulated by such equipment later in this book.

Of course, the number of dots per inch is also an important measure of a system's resolution. With flatbed, sheetfed, and most overhead scanners, the distance between the sensor and the material being scanned is fixed. In the case of flatbed and sheetfed scanners, this distance is only a fraction of an inch, so there is a 1:1 correlation between the number of sensors per inch and the resolution of the scanner. A sensor strip in a typical scanner that is 8.5 inches wide will have 2550 individual elements, or 300 per inch. There are also higher resolution scanners that boast 400 or more sensors per inch. These are more difficult and expensive to make, however, and they produce larger image files.

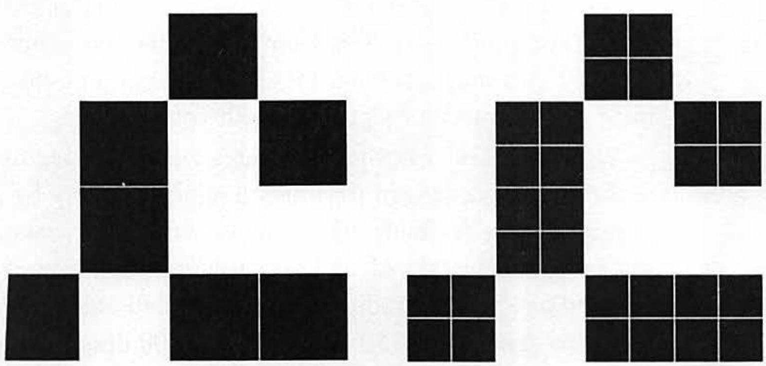
As you'll see later in this book, resolutions higher than 300 dpi may not do you much good. OCR programs function better at around 200 dpi, and you may not want to scan at resolutions that generate gray information you can't reproduce with your printer.

With scanners of this type, the area being scanned or the resolution of the image is changed from the maximum simply by ignoring some of information available. The scanner sensor may travel across only a few inches of the flatbed, and the software may ignore most of the data scanned on a line to produce a scan of a small area. By the same token, if you're scanning at 150 dpi instead of 300 dpi, the scanner software will interpret the information it receives to, in effect, ignore half the available data.

Many scanners will also interpolate the information they capture to produce image files that simulate higher resolutions. That may sound like magic, but the explanation is simple. The scanning software looks at the values of the pixels that surround each picture element in an image. It is then able to make some reasonably intelligent guesses about the smaller elements that would make up that pixel if it had been scanned at a higher resolution. You're not getting any additional information; the data that is captured is simply divided up into smaller pieces. That can be useful when an image is enlarged later on; the image may appear less chunky. Figure 2.11 shows a portion of a scanned image and that same image segment interpolated to a higher resolution.

The resolution of scanners that allow you to vary the distance between the subject and the sensor is a bit more complicated. Such scanners have an optical system that focuses the light on the sensor array. The sensor remains the same size, but the area of the original subject matter varies considerably. If the subject is placed very close to the sensor, it may capture an image area of only 1 x 1 inch. If the subject is moved farther from the sensor, the same sensor will capture a larger area—say, 10 x 10 inches.

Figure 2.11



A scanned image (left) and how your scanner may interpolate it to simulate higher resolutions (right).

You can see that simply moving the subject matter closer to this type of scanner will produce higher effective resolution. The Kodak 4 megapixel array would provide an effective 2048 dpi resolution when imaging a 1 x 1-inch subject but a much lower 204.8 dpi for a 10 x 10-inch original. The number of pixels captured in each case is the same—4,194,304—but in the latter example they are spread over a much larger area.

The scanning you do with a video scanner will probably have much lower resolution—typically around 350 x 300 pixels if you use a consumer camcorder and up to 525 x 400 pixels if you use a professional video system. But the same principles apply. Capture a 1 x 1.5-inch area with your camcorder, and you'll produce an image with about 300 dots per inch. Shoot a more typical image, and your resolution will drop between 30 and 50 dpi. Fortunately, those values are entirely practical for many applications.

Scanners that achieve resolutions even greater than those of the Kodak CCD array are available for very demanding work. These use special techniques to take a "snapshot" of the subject from several different positions. The information is combined to produce resolutions as high as 4096 x 4096 pixels.

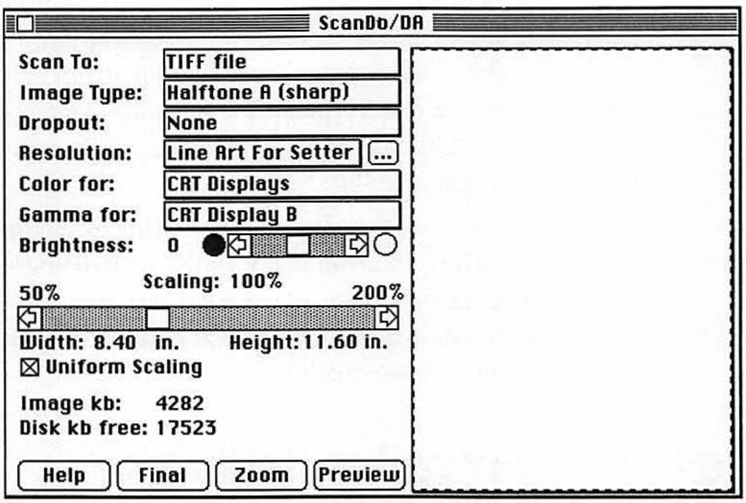
Scanner Controller Software

The software that drives a scanner may provide controlling functions only, or it may also have the ability to edit and manipulate the images you have scanned. Some of these functions will interface directly with your scanner hardware. For example, you may be able to control the brightness (threshold) of a scanned image and the distance the scanner covers (size of the image).

Other functions are entirely software-dependent and therefore will vary with the scanner controller software you choose. Some packages may allow only relatively large increments of resolution—say, 50, 75, 150, 200, and 300 dpi—while others will let you scan at any resolution from 25 to 600 dpi. Your scanner always scans at a fixed set of resolutions; the others are provided by the software.

You aren't limited to the controller software packaged with your scanner. Many of these programs are prepared as OEM products and sold directly by the developer to the scanner vendor. A few are also sold as aftermarket software directly to consumers. These programs offer some image manipulation features, but they aren't intended to be full featured packages. You may be surprised, however, at how much you can do with software like ScanDo, shown in Figure 2.12.

Figure 2.12



Scan-Do, a scanner controller package.

A much larger category of software includes the combination image-editing/scanner controller packages. These are intended primarily as gray scale or color editing programs. Scanner controlling functions are built in to add convenience; you don't have to exit your editing session to scan a new image. Such software usually has more capabilities than packages intended primarily to control scanners. By the same token, they may offer only limited control over your scanner.


#1
N.Y. TIMES
HARDCOVER
BESTSELLER
NOW ONLY

a whole raft of programs I can use to control my scanners, , ColorStudio, OmniPage, and Photoshop.

ıer manufacturers have been bundling larger amounts of third- oftware with their products in recent months. For example, Epson you ColorStudio, ImageStudio and ScanDo. The capabilities of packages overlap considerably.

lowever, everyone benefits from this arrangement. The software ndors provide the programs to the scanner manufacturers at an xtremely attractive price. The scanner companies are able to offer more ully functional products at little cost. You receive software that sells for up to $695 when not bundled with a scanner. The software vendor gets exposure to a rich trove of new users, who will certainly be tempted to upgrade as enhancements are made to their "free" software. A few upgrade fees more than compensate the software company for their cut-rate sales. Since bundled packages usually include drivers for only the hardware with which they are sold, there is little danger that the software will find its way onto the machines of other users. For example, while I have both Epson and HP scanners, I can't use the ColorStudio I received with the Epson with my HP scanner unless I get a copy of the regular package.

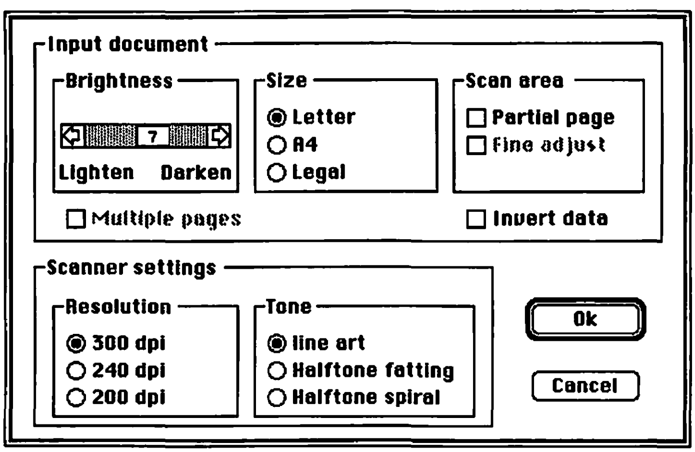
I'll explore image manipulation features later in this book. Right now, let's concentrate on the scanner controller aspects of this software. Figure 2.13 shows a typical scanner setup window.

- **Resolution.** Controller software lets you select from a range of scanning resolutions, from perhaps 30 dpi up to 600 dpi and beyond, depending on your scanner. You can select from fixed resolution increments, which can be as small as 1 dpi or as large as 25 or 50 dpi. The 1 dpi increments are obviously preferable, particularly if the software provides slider controls that can be used to adjust the resolution conveniently.

- **Scale factor.** If you know what size you want your finished image to be, you can scale it during the scanning step and save yourself some time. Scaling again may be accomplished using large fixed increments or smaller 1% steps. You may be able to scale from 25 to 100% and beyond.

- **Brightness.** This control adjusts the threshold the software uses to judge whether a tone is black or not. Raising the brightness is a lot like adjusting a television picture; extra brightness can boost some dark tones — at the cost of lighter tones at the other end of the scale. You can use this control to salvage pictures that have useful detail in areas that are either too dark or too light.
- **Contrast.** This is a measure of how widely the gray tones in your image are spread and is related to brightness. Contrast is usually adjusted in the same way as brightness within a given software package.

Figure 2.13



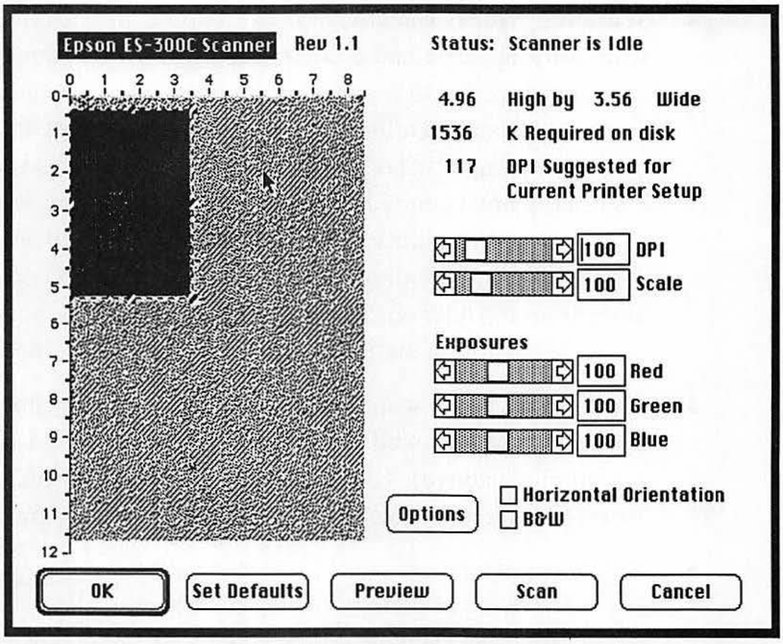
Scanner setup window

- **Scanning Mode.** Usually, you can choose line art, halftone, or gray scale (or gray scale and color). Line art is art that contains only black and white tones or that you want represented as a monochrome image. The halftone setting tells your software to process the image to simulate gray tones using black dots of various sizes. This is also called dithering. It's usually not a good idea to dither an image during the scanning step, as you lose the ability to resize your image and to perform other manipulations, including adjusting brightness and contrast. We'll explore dithering later on. Selecting the gray scale or color choice tells the software to capture the image with a full range of tones or hues.
- **Pixel depth.** Some scanning software also lets you choose from different pixel depths as well. Your choices may include 1-bit (the same as line art mode above), 4-bit, 8-bit, and 24-bit. The 4-bit and 8-bit modes are often offered as 16 colors/grays and 256 colors/grays, respectively.
- **Halftone patterns.** If you've selected halftone mode, you'll usually be offered a choice of patterns. These have names like Fatting, Spiral, and Bayer. You may even be able to choose the angle at which the dots are placed and the line frequency. Those considerations will all be explored in the section on halftoning later in this book.
- **Scanning area.** Most scanner software provides the ability to prescan the entire image. You can then view the preview and adjust the margins to the exact area you want scanned. This eliminates wasted image area and reduces the size of your files. Figure 2.14 shows a preview window.
- **Inverse image.** If you want a negative image, you may be able to invert it (reverse values of the tones in scanner parlance—not turn upside down) during the scanning step.

Summary

Most people won't need to know how a scanner works to use one. However, the information in this chapter can help you avoid some problems and provide the basis for understanding how to achieve better quality scans.

Figure 2.14



A preview window, which allows you to specify exactly the area to be scanned. If you scan only the area you are interested in, you'll save disk space.

There are five basic types of scanners: flatbed, sheetfed, overhead, hand-held, and video. All five use a sensor of some type to capture images reflected from or transmitted through the original by a light source. The sensor, usually a charge-coupled device (CCD), is made up of thousands of individual elements, each of which can capture a single picture element, or pixel. If very little light reaches the sensor, the pixel is stored as a dark tone. If a great deal of light makes its way to the sensor, the pixel is represented by a light tone.

Linear array scanners capture one line of pixels at a time in a process that is repeated until the entire subject is scanned, line by line. On a flatbed scanner, the original is placed face down on a piece of glass, and the sensor or a mirror that reflects light to the sensor is moved underneath it. With sheetfed scanners, the sensor and the light source remain fixed, and the original is pulled past them using a roller transport system.

Hand-held scanners use your muscles to pull the scanning head past the material being scanned. Overhead scanners place the sensor above the copyboard that holds the original and capture the image that way. Some overhead scanners and all video scanners use a two-dimensional array of sensors to capture an entire original image at one time.

If the scanner is a color model, it must have either three individual colored light sources, one for each of the primary colors, or a system of filters that can be placed over the light or the sensor.

The information scanners capture is converted from an analog value to a digital value before it is stored. The number of bits available to store this information determines the pixel depth of a scanner. A 1-bit scanner can store only black and white information. A 4-bit scanner can capture up to 16 gray tones per pixel. Full 8-bit scanners may record as many as 256 tones. Color scanners capture 8 bits of information for each of the three primary colors and therefore record 24-bit information.

If an image has large areas of tone that change gradually from one to another, you need more shades of gray to provide a smooth transition. Otherwise, there will be abrupt changes from one tone to the next, producing the banding effect that is called posterization.

Resolution is the amount of information per linear inch or square inch that can be captured or reproduced by a system. It includes pixel depth as well as dots per inch.

With most scanners, the area being scanned or the resolution of the image is changed from the maximum by ignoring some of the available information. Many scanners also interpolate the information they capture to produce image files that simulate higher resolutions.

Video scanners allow you to vary the distance between the subject and the sensor so resolution can vary. If the subject is placed very close to the sensor, it may image an area only 1 x 1 inch. With the subject moved farther from the sensor, the same sensor will capture a larger area, say, 10 x 10 inches. The number of pixels captured in each case are the same, but in the latter example they are spread over a much larger area.

While you receive scanner controller software with your scanner, you can also select third-party offerings that may have some additional features.

These packages may provide controlling functions only, or they may also have the ability to edit and manipulate the images you've scanned.

The following are some of the controls you need to know about:

- Resolution.
- Scale factor.
- Brightness.
- Contrast.
- Scanning Mode.
- Pixel depth.
- Halftone patterns.
- Scanning area.
- Inverse image.

3

Scanners and Memory

The demands that scanners make on permanent magnetic storage (your hard disks and other media) and on temporary storage (random access memory, or RAM) deserve some special attention. It's easy enough to say that the more hard disk space and RAM you have, the better, and let it go at that. But the technology of both sorts of storage is changing so rapidly that you need more specific information. How much storage is too little? How much is a lot? How do you use it once you have it?

This chapter looks at memory as it applies to scanners, while the following chapter examines disk storage. Both discuss the demands that scanners make on these resources and explain how you can get the most from what you have.

I'll be going into a bit of detail in the section on memory, because I've found that this area holds considerable potential for confusion and problems for scanner owners.

The magnetic storage section isn't very technical at all aside from the discussion of data compression technology; you don't have to understand how hard disk drives work to select your best permanent storage option.

Memory Is Made of This

As someone who was alive at the dawn of personal computing, I find today's cavalier attitude toward disk and memory capacities quite incredible. My first desktop computer had 4K of memory, which itself seemed rather lavish to those who built their PCs from kits in the mid-1970's. I once paid $300 for a 16K upgrade and used a battery of four 80K floppy disk drives to store programs and data. I had been using a cassette recorder that took as long as 10 minutes to load a program.

Things seemed to improve quite a bit when I got my first Mac, an original 128K model, in 1984. Here was a massive amount of memory, so much that I scarcely saw how I was going to use it all. A few years and a thousand disk swaps later, I discovered just how easy it is to eat up large amounts of memory with RAM disks, RAM caches, font caches, INITs, loads of desk accessories and—of course—scanned images. By that time, though, I'd upgraded to a 512K "Fat Mac," which I later switched for a Mac Plus. For several more years, I got along quite happily with 1 Mb of memory, an internal 800K disk drive, one external 800K drive, and no hard disk at all.

When I started scanning images in earnest, even a megabyte of RAM wasn't enough, so I learned the mysteries of the Torx screwdriver and SIMMs, and upgraded my Plus with 4 megabytes worth of chips. I also purchased an 80 Mb external hard disk to store images.

That worked fine until I made the final jump to a Mac II with 24-bit color, and discovered there were programs like ColorStudio that actually needed 5 Mb to run under MultiFinder, and worked even better with 8 Mb of memory. That's where my memory chase ended, since my original Mac II couldn't handle anything more than 8 Mb of RAM, barring virtual memory techniques. (More on virtual memory later.)

As you'll see in the following sections, a little extra memory or hard disk storage may be one of the best investments you can make toward improving your scanner's performance and capabilities.

Scanners and RAM

Scanned images are big. If you're new to scanning, this may not have sunk in yet, but remember that when a single byte is used to represent 256 different gray levels for each pixel in an image, you need as much as 8.4Mb to store one 8.5 x 11-inch image. Increasing the resolution of the scanner to only 400 dpi almost doubles the size of the image to

15Mb. Full color images are three times larger. The very largest are so big that no scanning or image editing program could possibly keep an entire image in memory at once. Instead, only pieces of the image are loaded into memory, and the rest stored on hard disk. That provides you with a form of virtual memory; your hard disk simulates memory for the purposes of the image editing program.

Fortunately, most of the scanned images we work with are considerably smaller. Of the thousands of images I've scanned, only a small percentage took up a full page. These were generally 8 x 10-inch photographs that were to be reproduced at smaller sizes. When you're working with an orginal that large, and intend to reduce it, you'll rarely need to scan at resolutions higher than about 75 dpi.

As I noted, memory hasn't always been cheap. In 1988, I paid a discounted price of $895 for 2Mb of add-on memory. I considered myself very lucky, because memory prices doubled shortly thereafter. High memory prices of the past are part of the reason why the Mac II, introduced in 1987, is capable of handling only 8 Mb of memory. That limitation may have made sense at the time; after all, who can afford more than, say, $8000 worth of memory for a desktop computer?

Unfortunately, that short sightedness affects us today when RAM can cost as little as $40 a megabyte. Eight megabytes is only $320 worth of cost-effective memory.

With prices this low, if you're heavily involved in scanning there's almost no excuse not to have as much memory as your computer can handle, whether it's 4, 5, 8Mb, or some other figure. Unless you're using a hand-held scanner, a few extra megabytes of RAM are likely to cost only a tiny fraction of the total price of your system.

The RAM situation will only improve in the future, since extra RAM itself has become a more attractive feature, and has become essential for running System 7. There are a lot more things you can do with it. You can use extra RAM as a fast virtual disk to speed up operations. Some of it can be set aside for a caching program, or as a font cache for Adobe Type Manager. RAM is handy for voice input, animation, and multimedia.

Under MultiFinder or System 7, you can load a scanner control program in one window, operate an image manipulation package like Digital Darkroom, and run Ventura Publisher or PageMaker in a third. Given enough memory, all three can be operating at the same time. As I noted, virtual memory is a way of making some of your free hard disk space mimic actual RAM, allowing you to load more programs than will fit in the memory you have installed in your computer.

Upgrading to 4Mb or 8Mb of RAM will become as common as moving up to white-wall tires. Moreover, 4 megabit chips are now available for approximately $125 a megabyte. Another price reduction of about 50% will mean that some of us who are limited to 8Mb now can replace that with 32Mb of memory at reasonable cost.

Memory is relatively cheap today and should become even less expensive in the future.

Memory Is Made of This

Your Macintosh addresses its memory through a set of parallel wires called address lines. The original Motorola 68000 chip which is used in the Mac Plus and earlier Macintoshes, as well as the Mac SE and Classic, has 24 address lines available. All other Macintoshes use the 68020 or 68030 microprocessor, which has 32 address lines available. Theoretically, 68000-based Macs could use up to 111111111111111111111111 bits of memory (in binary; that's 24 1's), or 16 megabytes of memory, while other Macs could access up to 4 gigabytes of RAM.

In practice, the actual usable amount is much less, for several different reasons. The Plus, SE, and Classic all have just four sockets (the Classic has just one, and requires an add-on card to gain the other three) for single in-line memory modules (SIMMs), limiting you to 4 Mb when you use 1 Mb SIMMs. The ROMS in these machines won't recognize 4 Mb SIMMs, so your upper limit for any of these is a paltry 4 Mb.

Other Macs have two to eight sockets, allowing anywhere from 8 Mb to 128 Mb (in theory), depending on what kind of SIMMs you use. The

Mac II, IIx, and IIcx top out at 8 Mb, while the IIsi and IIci allow 17 and 32 Mb, respectively. The top of the line Macintosh IIfx can accept 128 Mb of RAM, if you use 16 Mb SIMMs, and have the wherewithal to afford them!

Of course, System software prior to System 7 limits your Mac to using 8 Mb under (almost) any circumstances. That's because of the way in which the memory address space was allocated. With 24-bit addressing, a total of 16 Mb of memory can be accessed. Apple allocated only the first 8 Mb for RAM. The next megabyte was assigned to ROM, then the 6 Mb after that set aside for the NuBus slots (one megabyte for each of up to six slots). The final megabyte was reserved for I/O operations.

Special virtual disk programs (such as Virtual, by Connectix) can use the paged memory-management unit (PMMU) chip, the 68851, available for 68020-based Macs, or the built-in memory management capabilities of the 68030 and 68040 machines. Such software borrows address space from the NuBus. You can then use available hard disk storage as if it were memory, up to the memory-handling limits of your system. Magnetic memory is typically much slower than the silicon kind, but virtual memory techniques can help you run programs and combinations of programs under MultiFinder (or System 7) that you couldn't use otherwise.

Another Connectix offering, Maxima, also uses the PMMU or the capabilities of 68030/40 chips. It allows Mac models IIfx, IIci, IIx, IIcx, SE/30, or earlier Mac II's with a PMMU installed to use up to 14 Mb of actual RAM. The extra memory chips can be added either by using 4 Mb SIMMs in your existing slots, through multimegabyte composite SIMMs, or by installing a NuBus RAM card.

If your computer is an SE/30, which doesn't have a NuBus, a full 6 Mb of additional RAM becomes available. You must subtract 1 Mb for each NuBus card you have installed. For example, a Mac II with PMMU and a 24-bit color video card installed can use 5 Mb of additional memory, for a total of 13 Mb.

Of course, you can't install memory in the precise increments that Virtual will allow you to use. You might have 16 or 20 Mb installed, with a maximum of 14 Mb available as ordinary RAM. The rest doesn't go to waste. Maxima will use it as a RAM disk, which you can use as very fast storage for programs, temporary files, or (if you're brave) data. Just remember to copy any files you want to keep from the RAM disk before you shut down your computer. Maxima lets you specify particular files which should be copied to the RAM disk when you start up the Mac, and whether modified files should be copied back to the hard disk on shutdown.

Without utilities of this sort, even with System 7, you'll need "32-bit clean" ROMs to crack the 8 Mb barrier. These are found on later Macs: the Mac LC, IIci, IIsi and IIfx at this writing. With these machines, a 32-bit addressing scheme can be used that allows each of them to address up to 128Mb of physical RAM and, with all but the LC, up to a gigabyte of virtual memory. Note that not all applications are compatible with 32-bit addressing; with these you'll be limited to the memory that can be accessed using 24-bit addressing.

Getting the Most from Your Memory

I realize that my discussion of cheap memory in the first section of this chapter provides little comfort for those on a restricted budget. Many people use scanners in businesses that ought to be able to easily cost-justify the expenditure of a few hundred dollars when the payback is so immediate and dramatic. However, I recognize that smaller businesses and individuals who want to use scanners may not be able to toss off money in $100 increments. Fortunately, there are some things you can do to stretch the memory you do have. I mentioned some of these applications in the previous section.

- Make sure your memory is the fastest you can use. If your computer requires 80 ns RAM and you load it up with 100 ns memory, performance suffers. Today, the difference in price between the fastest memory and the slowest is likely to be only a few cents per chip. Don't skimp.

  Actually, memory has gotten so inexpensive that some dealers don't bother to stock slower chips. I called a mail order firm to buy 120 ns memory for a slower computer I have, and the dealer tried to sell me up to 80 ns chips for the $10 per SIMM difference. Hey, $10 adds up to $80 when you're buying eight SIMMs, so I stuck to my guns. The dealer finally revealed that he didn't stock 120 ns SIMMs anymore and sold me the 80 ns versions for the lower price. The faster SIMMs won't run any faster in my particular computer and pulling them is such a pain that I'll never bother to reinstall them in a faster computer, so I really didn't gain anything.

You might wonder who would be foolish enough to try to get by with slower RAM. You might if you didn't know the speed of the memory you had or needed or what speed memory would be of use to you.

You should have little difficulty determining what type of memory to add to your motherboard. Macintoshes earlier than the Mac SE (this includes the Mac Plus) required 150 ns RAM. Early Mac IIs, like the II, IIx, and IIcx, along with the SE/30 and Mac LC need 120 ns RAM. Only the Mac IIsi and Portable use 100 ns RAM; the Mac IIci and IIfx require the fastest 80 ns SIMMs. Standard, low-profile SIMMs work in all the Macs but the IIfx, which requires special 64-pin units that are like those used in the LaserWriter IINTX.

The memory chips themselves are marked with a code number that ends with a dash and a number that represents the speed of the chip: -12 means 120 ns memory, -1 is 100 ns memory, -8 is 80 ns memory, and so forth.

- Create and use a RAM disk. If your image-editing program is an older one, it may not use all the RAM you have available. In such cases, you can gain some speed by setting up a RAM disk and using that to store your images while you edit them.

- Pay attention to how your System file is using RAM. Those INITs use up valuable memory; do you really *need* Oscar the Grouch telling you how much he loves trash every time you discard a file? Pare your INITs down to the minimum, or use a utility like INITpicker to choose the ones you want at start-up time.

  Fonts and desk accessories also use RAM by enlarging the size of your System file. Again, there are utilities like Suitcase II that can make all your DAs and fonts available to you without storing them permanently in memory that could be used for scanning or image editing.

System 7 changed the way fonts and desk accessories are manipulated. You no longer need Font/DA Mover. Fonts can be installed by dragging their icons into the System Folder. Desk accessories have become ordinary launchable applications. You can place them in any folder on your disk, or into the Apple menu (which can now include things other than desk accessories.)

- Caching programs, particularly the font caching used by Adobe Type Manager, can improve performance, but also take up RAM. Make sure you've balanced your need for free memory with the performance you require. You may find that for image editing sessions, you don't need ATM at all, and can disable its cache to free up extra RAM for your work session.

  Memory cache cards, which are different from accellerators, and also different from the caches available from your control panel, can also improve performance. Your control panel caches allocate some of your system RAM to storing information loaded from your disk drive. In the case of the main cache, this can be any sort of program information or data. The font cache keeps font bit-map information in memory, to allow drawing screen fonts quickly without needing to load the data from disk each time. Both types of caches use up valuable system RAM.

  Add-on caches perform a different function. Cache cards use faster, 25 nanosecond static RAM in place of dynamic RAM, which usually has no better than an 80 ns rating. This quicker static RAM is used to store frequently used data and instructions, and supply them to the CPU as fast as the microprocessor can handle them. Just as a disk or font cache keeps your system from slowing down waiting for information stored on disk, a memory cache eliminates waits for information from slower RAM chips by providing a holding area populated with much faster RAM.

  As a bonus, cache systems of this sort don't rob you of free RAM. The Mac IIfx has such a cache built in, while the IIci has a special slot which can be used either for an accellerator (costing up to thousands of dollars) or a cache card ($200-$300). The cache card provides a significant speed-up for many operations at a bargain price.

Cache cards will also soon be available for other Macintosh models. While there are no such products for the Mac LC or SE/30 at this writing, expect some in the near future. The Mac IIsi can use special cache modules which plug into the direct-slot adapter. Those of you with Macintoshes equiped with the 68000 microprocessor, including the Mac Classic, don't need a cache add-on. The bottleneck in your systems is the microprocessor – not the RAM chips.

Summary

This chapter that explored just why memory and magnetic storage are critical to your happiness as a scanner user. While it's easy to say that the more RAM and hard disk space you have, the better, the issues are considerably more complicated than that. This chapter shows you how much storage is too little, how much is a lot, and what you can do with it once you have it.

Understanding memory is essential because some scanner software absolutely requires certain amounts.

Now that memory has become so cheap, there are ways you can get more from the memory you have using memory managers, RAM disks, faster RAM chips, and more sophisticated graphics cards with additional memory.

Scanned images are big. An 8.5 x 11-inch image at 400 dpi requires 15Mb. Full color images are three times larger. The very largest are so big that no scanning or image editing program could possibly keep an entire image in memory at once. Instead, only pieces of the image are loaded into memory, and the rest stored on hard disk. That provides you with a form of virtual memory; your hard disk simulates memory for the purposes of the image editing program. Fortunately, most of the scanned images we work with are considerably smaller.

With prices this low, if you're heavily involved in scanning there's almost no excuse not to have as much memory as your computer can handle.

Under MultiFinder or System 7, you can load a scanner control program in one window, operate an image manipulation package like Digital Darkroom, and run Ventura Publisher or PageMaker in a third. Given enough memory, all three can be operating at the same time.

The original Motorola 68000 chip which is used in the Mac Plus and earlier Macintoshes, as well as the Mac SE and Classic, can address 16 megabytes of memory, while other Macs can access up to 4 gigabytes of RAM.

In practice, the actual usable amount is much less. The Plus, SE, and Classic all have just four sockets (the Classic has just one, and requires an add-on card to gain the other three) for single in-line memory modules (SIMMs), limiting you to 4 Mb when you use 1 Mb SIMMs. Other Macs have two to eight sockets, allowing anywhere from 8 Mb to 128 Mb, depending on what kind of SIMMs you use.Of course, System software prior to System 7 limits your Mac to using 8 Mb under any circumstances, unless you have a special virtual disk or RAM disk program and a paged memory-management unit (PMMU) chip required with microprocessors prior to the 68030.

To get the most from the RAM you have:

- Make sure your memory is the fastest you can use.
- Create and use a RAM disk. If your image-editing program is an older one, it may not use all the RAM you have available. In such cases, you can gain some speed by setting up a RAM disk and using that to store your images while you edit them.
- Pay attention to how your System file is using RAM. Those INITs, fonts, and desk accessories also use RAM by enlarging the size of your System file. Again, there are utilities like Suitcase II that can make all your DAs and fonts available to you without storing them permanently in memory that could be used for scanning or image editing.
- Caching programs, including the font cache used by Adobe Type Manager, provide extra performance at a cost of free RAM. Add-on cache cards can speed up the Mac IIci, which has a special slot on the motherboard for this accessory.

4

Scanners And Disk Storage

Although scanned images may occupy a great deal of memory while you are working on them, you can use that memory over and over. You need only as much memory as the largest image you plan to work with demands. However, once you start storing images, you'll awaken an insatiable appetite for hard disk space or alternative storage media. Images you want to save permanently quickly begin devouring your hard disk storage.

You can almost tell what type of work a person does by the size of his or her hard disk. Someone who is content with a 40Mb or smaller hard disk is probably doing word processing, working with spreadsheets, rating insurance policies, or involved in some other character-based task. In the offices I visit, the disk drives of such computers rarely house more than 15 to 20Mb worth of applications. All the data files associated with those programs usually consume another 10 to 20Mb.

Today, a 40Mb hard disk is considered standard for most users, although you'll see an increasing number of 65 to 80Mb drives as the cost premium associated with the higher capacity units drops to almost nothing. Hard disks for Macintoshes have dropped significantly in recent months; an 80Mb drive that once cost $800 or so may be priced at $500 today.

Whenever I find an individual workstation (not a network server) with a 300Mb or larger disk, it's an even bet that some heavy duty image work is being done there. Data files that large are probably too valuable to reside on a computer that serves only one user. But 300Mb is nothing at all in terms of images. A decent-sized bit-mapped clip art collection can eat up that much storage in no time at all.

No scanner user is going to be happy with 40Mb for very long. If you must make do with that little disk space, plan on archiving many of your scanned images to floppy disks or some other medium. High density 1.44Mb 3.5-inch disks are preferred to 800K disks. Just remember that even a high density disk may be unable to store a single very large scanned image and may be limited to only a few average-size gray scale images. Line art is much more compact and can be archived to floppies efficiently.

In general, on a per megabyte basis, hard disk storage becomes less expensive as the size of the disk increases. A 40Mb drive may cost $400, and an 80Mb drive with twice the capacity only $500, while a 330Mb drive can be purchased for $1400 or less. I've even seen refurbished 675Mb drives advertised for $1395 in recent months.

If you can afford an 80Mb hard disk, that ought to be your minimum for scanner applications. An 80Mb drive gives you enough room for all your software and a good complement of scanned images. You'll also have some extra disk space for programs that use virtual memory techniques to simulate extra RAM by storing pieces of images on the hard disk.

Making Your Hard Disk Go Farther

Since any hard disk will eventually be too small, you should be aware of the various things you can do to make the disk storage you have go farther. In this section, I'll tell you a little about image compression.

Actually, you may find a bit more than you wanted to know about this topic, but image compression is an important technology to users of scanned images, so I want to explain it in some detail. I've simplified the topic considerably: you won't find any discussions of discrete cosine transformation as it applies to image compression, since I don't understand the math myself. However, you may have more detail than you need for everyday scanner use. If you're curious or willing to follow a detailed explanation, read on.

One way to fit more scanned images on your hard disk is to use an image compression scheme. Many image editing packages offer the option of compressing files automatically when they are stored, usually using an industry standard algorithm. The images are automatically decompressed when they are reloaded. You may save as much as 50% of the disk space that would have been required. The only penalty is the increased time required to store and retrieve compressed files. Also, you may find that some packages don't read files that have been compressed by other software. ColorStudio's RIFF format, for example, can't be accessed by most other software, other than programs offered by the same vendor, LetraSet, such as DesignStudio.

When you're using a program with a proprietary image format, you should remember to save the original image in uncompressed format or a more common format, such as PICT or TIFF.

Image Compression

At this point, you may be wondering what image compression is and why it isn't done automatically. As I warned, we'll digress for a moment to explain some of the concepts behind this important tool for scanner users.

When a computer is working with a program or a scanned image, it stores a binary number representing each piece of program code or part of the bit map in a separate memory location. In the case of programs, the numbers stand for instructions for the microprocessor, memory locations, or text strings. In an image file, numbers represent values in the bit map and other information.

For example, if the computer encountered the two binary numbers 10110100 and 00001001 in a machine language program, it would interpret them to mean a particular set of instructions, such as to load a value of 9 decimal, or 1001 in binary, into a special memory location on the microprocessor.

The same two numbers in the middle of a 256-gray scale bit map would mean something quite different. The 10110100, which is 180 in decimal, might represent a fairly dark gray pixel (since it has a value of 180 on a scale of 0 to 255). The 00001001, on the other hand would represent a very light, almost white, tone. In actual scanned images, however, you'd rarely encounter two adjacent pixels with such divergent values. And that helps explain why different compression methods must be used for programs and data than for image files.

Data Compression vs. Image Compression

All data compression schemes operate by replacing individual streams of bits with shorter streams that convey the same information. When squeezing a program, the compression/decompression process cannot lose a single bit of information. Otherwise, the data or program becomes corrupted and no longer useful for its intended purpose. Only a conservative, so-called lossless compression scheme can be used for these files.

Single bits are rarely so important in image files. An 8 x 10-inch image scanned at 300 dots per inch in binary, black-and-white mode contains 7.2 million individual bits. If some of the black pixels are displayed or printed as white, or vice versa, you may not even be able to detect the difference visually, as long as they aren't clumped together to produce a larger white or black spot on the image. Such an image probably contains large spaces of the same value, as when black line art is drawn on a white background. Most of the file consists of binary 0s interrupted by a few strings of 1s here and there.

If a portion of a given line of the image looks like this:

00000000000000111000000000000000111111000000000000
0000 000000001111111111111111

why would you want to use up nine whole bytes to store the information about those 72 consecutive pixels? A simple way to convey the same data would be with a string of numbers like this, which takes up only 4.5 bytes:

001110000011010000000110001100010000

If you divide that string into sections of six bits each and then translate the binary numbers into decimal, you get the following series 14, 3, 16, 6, 24, 16. That happens to represent the length of each string of consecutive bits of the same value in the original number above. That is, the line begins with 14 white pixels, followed by 3 black pixels, then 16 white pixels, 6 black, 24 white, and 16 black. What I've done here is allocate a run of six bits to store a value that represents the number of pixels in a row for either black or white.

Since there are only two possible states for each pixel, we don't need to convey any information about what type they are, just the length of the run of one type before we need to switch over to the other; that is, only the transition points need to be conveyed. Each group of six bits conveys a number from 0 to 63, so we can represent runs of up to 64 individual pixels in either black or white.

As noted above, in 6-bit binary chunks that second string of 0s and 1s translates into 14, 3, 16, 6, 24, and 16. Runs longer than 64, if they occur, can be designated by using two or more sets of 6 bits. The success of my simplified scheme hinges on most of these runs being at least 6 bits and less than 64 bits long on average. Runs of fewer than 6 bits can be more efficiently represented by the unencoded binary string, naturally, of 5 or fewer bits. Runs of longer than 64 bits require 12 bits to represent.

You can see that I've compressed the size of this sample line by 50% without losing any information at all. Actual compression algorithms, while considerably more sophisticated, are based on the same principle.

Figure 4.1 on the next page provides a simplified visual illustration of how some of the most common compression schemes work. You'll find it an easy way to absorb some of these concepts.

Figure 4.1

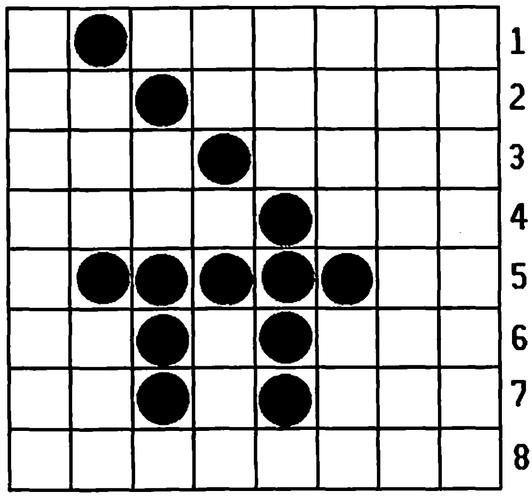


Image Compression Made Small

The illustration above shows a simplified bit map, measuring only 8 pixels by 8 lines. Actual scanned images, of course, have anywhere from 75 to 300 pixels per inch on every line, and an equal number of lines per inch. However, image compression concepts apply no matter how complex the image is.

Line 1 consists of one black pixel and seven white pixels. In many types of scanned images, particularly text, most of the pixels will, in fact be white. Instead of storing a value for every pixel on the line, our software can instead record the transitions from white to black to white again. So, instead of the binary string 01000000, which represents the state of each pixel in the line, we could store values representing only the numbers 1 and 1. That's because the line is assumed to start out with a white pixel (the most frequent occurence), and white pixels continue for a run of just one (if the line had started out with a black pixel, we could store a 0 first instead). Then, we have a run of one black pixel, after which the line switches back to white again for the rest of the line. The two 1's give us everything we need to know to reproduce that line.

The second line is just the same as the first, except that the single black pixel is shifted over one space. This happens to be a common thing, especially with images that include diagonal lines. Some compression schemes allows storing a short code that means exactly that.

Lines 3 and 4 are more of the same. A single code might indicate that the next three lines after line 1 should be stored as the same line, just shifted over one pixel each time.

Line 5 is different, and can be effectively stored just by storing the figures representing the transitions again (1 and 5). Lines 6 and 7 are identical, which is often the case in scanned images, so a single code that means "same as last line" can do the job. Line 8 is a blank line, and requires only a blank line code.

Actual compression schemes are considerably more sophisticated than this, but you should understand the general idea from this illustration.

Gray Scales and Image Compression

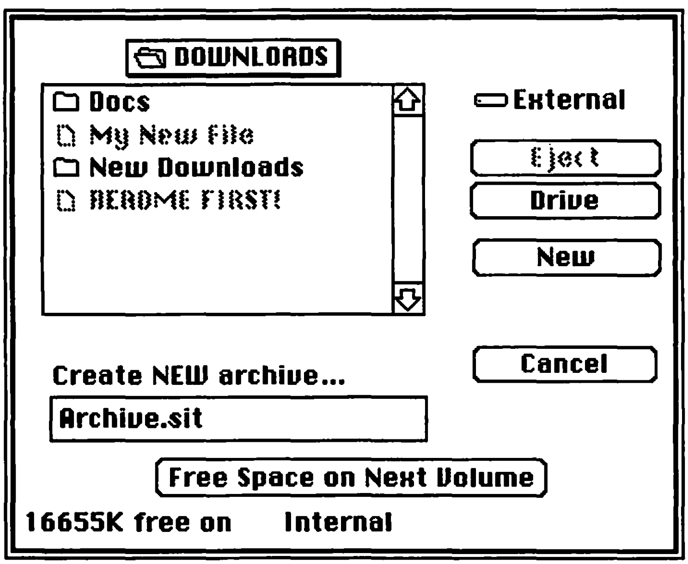
Gray scale images also contain strings of repeating numbers, but they are more complex to encode, because up to 256 different values may be associated with a given pixel. However, gray scale images rarely change abruptly from one tone to another. There are long stretches of pixels that have the same or similar values, and gradients are often very predictable. Image encoding schemes can take this predictability into account.

Some information may, in fact, be lost using these more aggressive algorithms. A gradient in the uncompressed version may not change from gray level 210 to 211 at the precise point it changes in the original image. But, at a resolution of 300 dpi, are you likely to be able to notice?

Of course not. That is why compression systems used for images can be more efficient than those used for software. Minor image degradation is likely to be undetectable. Such compression schemes are known as, so help me, lossy, to differentiate them from lossless systems.

A lossless compression package, such as STUFFIT or STUFFIT Deluxe, may not provide as much compression as you can use with image files. You are probably better off using the compression system built in to your scanner or image editing software. In fact, I tried squeezing a compressed TIFF file using one of the popular file compression programs and ended up with a file that was 39 bytes longer than the original. STUFFIT is shown in Figure 4.2.

Figure 4.2



STUFFIT, a shareware version of a popular archiving utility. The commercial software is called STUFFIT DELUXE, and there is also a freeware program, UNSTUFFIT, which can be used only to extract files.

The only real reason to use such programs with compressed images is to build an archive of image files that can be handled as one convenient package. For example, if I were going to transmit a group of small image files over a modem and didn't have a telecommunications protocol that could handle batch transmissions at the other end, I would probably load all of the files into a single archive file with no further compression. If the files were large, I'd probably send them one at a time. Anytime you transmit files that are larger than they need to be, you risk having your transmission interrupted and being forced to start over from the beginning. Not all telecommunications protocols let you resume an interrupted transmission from the point at which you left off.

Special stand-alone image compression software and hardware is already available for Macintosh color systems, chiefly because color files are so huge and problematic and because the Mac world is quite a bit ahead of the PC community in color capabilities. These products include Kodak's Colorsqueeze product, which offers three levels of compression and can squeeze images up to 64 times.

C-Cube Microsystems has put its CL550 image processor chip, which is based on a JPEG algorithm onto a $995 add-in board for both IBM PCs and Macs. This board reportedly compresses images by as much as 75 to 1. At last report, however, the company was phasing out direct sales to end users and was concentrating on selling and licensing its technology to OEMs.

Other novel solutions allow you to set up a hard disk volume as a special "squeezed" area. Any files stored to this portion of your hard disk are automatically compressed and then decompressed when they are retrieved. The process is said to be transparent and does not slow down hard disk access appreciably. Again, this is a lossless system that may not be the most efficient way to compress image files but can be effective in stretching your hard disk space a little further.

Compression Methods

There are several ways to compress the numbers used to represent images. One method is called Huffman encoding, in which the most frequently occurring numbers are represented by relatively short codes. Overall, if binary numbers, such as 11111111, which require the full 8 bits of a byte, occur frequently enough, they will be represented by much shorter binary numbers. The most efficient Huffman encoding schemes prepare a special frequency table for each file being processed, assigning the shortest codes to the numbers that occur most often. The table below shows how part of such a table might look:

Original number	Code
10111111	000
10010111	001
11011001	010
11101111	011
01110111	100
00111011	110
10000111	111

In this example, the seven most common values, which would require 53 bits to store unencoded, can be represented by only 21 bits. That's a saving of about 60%. With text files, compression can be impressive, because certain characters—e, t, a, o, i, and n, for example—occur much more frequently than others in our language. Images are less susceptible to compression using unadorned Huffman encoding. However, it can be combined with other methods that reduce images to a series of numbers which can themselves be further shortened using the Huffman method.

Lempel-Ziv (LZ) or Lempel-Ziv-Welch (LZW) compression replaces frequently used strings of numbers with fixed-length codes. This system uses statistical analysis to determine what sets of numbers appear most often.

Fax machines offer a good illustration of another compression method, in which entire lines of an image are analyzed. An entire blank line of, say, 1700 pixels can be represented by a single code. If several blank lines in a row are present, another code can signify that. Still another can indicate that the line to follow is the same as the previous line.

Within a given line, only the transitions from black to white need to be communicated. Color and gray scale images, of course, can't be compressed using the fax method, since there are no simple black/white transitions and few lines that are all one tone or exactly the same as the previous line.

JPEG

The Joint Photographic Experts Group (JPEG) has developed a compression scheme for continuous tone images that is efficient, and still retains the valuable image information. JPEG uses three different algorithms—one called discrete cosine transformation (DCT), a quantization routine, and a numeric compression method like Huffman encoding.

JPEG first divides an image into larger cells—say 8 x 8 pixels—and performs a discrete cosine transformation on the information. This mathematical mumbo-jumbo simply analyzes the pixels in the 64-pixel cell and looks for similarities. Redundant pixels—those that have the same value as those around them—are discarded.

Next, quantization occurs, which causes some of the pixels that are nearly white to be represented as all-white. Then the gray scale and color information is compressed by recording the differences in tone from one pixel to the next, and the resulting string of numbers is encoded using a combination of the Huffman and fax schemes. An 8 x 8 block with 24 bits of information per pixel (192 bytes) can often be squeezed down to 10 to 13 or fewer bytes. JPEG allows specifying various compression ratios, in which case larger amounts of information are discarded to produce higher compression ratios.

I used Kodak's ColorSqueeze software, a JPEG-compliant compression program, to squeeze down a 3Mb portrait using the highest compression ratio. The picture happened to have large areas of similar color without a lot of detail. The entire image compressed down to about 87K—about a 97% ratio! But when I restored the picture, the large tonal areas were quantized into unattractive blocks. This was a case in which JPEG wasn't suited to the particular subject matter at the highest compression ratio. A more detailed image at the same ratio wouldn't have been compressed so far and would have looked better on restoration. The same image at a lower compression ratio looked much better.

The moral is that you must be careful in using compression schemes on valuable images when little or no loss of detail is important.

Open Ended Storage

I can guarantee that your hard disk storage requirements will expand to fill your available hard disk space. Several months ago I was alarmed because I had only 30Mb free on my 200Mb hard disk system. I upgraded to a 330Mb drive and filled up a little more than half of it with the files from my old hard disk.

I'm not sure where the extra 50Mb of files now squeezed onto that disk came from. I know I picked up a few new applications while researching this book. Most of that space, however, was eaten up by image files that I no longer feel pressed to archive to other media now that I have all this extra room on my hard disk—or had all that extra room.

Unless you work with color extensively or really need to have vast image files available for near-instant access, I really don't recommend getting a hard disk larger than 330Mb. You eventually reach a point of diminishing returns.

Options

As I've already noted, ordinary floppy disks won't help much. They can barely store one large image file. Tape backup systems are an excellent choice for archiving image files you won't need for a while and for protecting yourself against hard disk crashes.

However, tape is a serial medium and best suited for capturing a mirror image of your hard disk. Tape systems can be used for incremental backups, and most have directories that allow you to restore only a particular file. Even so, retrieving that file may force you to read the entire length of a tape at times to find a single file that you want. The process can be very slow. Tape back up is the absolute champion in terms of cost per megabyte, however. Tape cartridges are so cheap that you may want to back up each file twice—particularly if you've ever had problems restoring a valuable file from tape. I do, however, think there are other solutions that are preferable if you don't need to pinch pennies, or you find yourself wanting to back up 100Mb or more to a single cartridge.

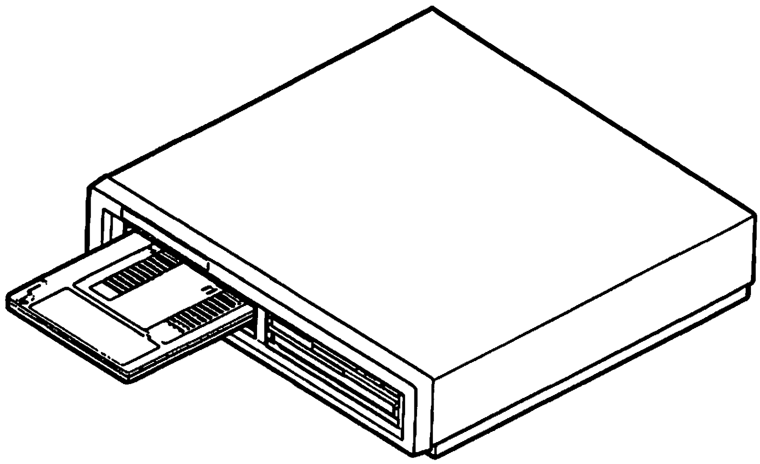
Removable hard disks are another solution whose time will never come, in my opinion. Even the most rugged removable hard disk media suffer from rough treatment, and they carry the highest cost per megabyte of any storage device. Removable hard disks are useful only for high security applications, since they allow you to pull the entire disk out and lock it up when not in use. New Yorkers who have removable stereo radios in their cars obviously took inspiration from this technology. They are already posting signs on their computers, "Hard Disk Already Stolen," when they leave their offices at night.

Other Removable Media

For convenience, flexibility, and reasonable cost per megabyte, there is currently only one option: removable media like Iomega's Bernoulli Box. Iomega's offerings have recently been improved, upgraded in features, and reduced in cost. This is the closest thing to an outright endorsement of a product you'll find in this book.

A Bernoulli Box is a SCSI disk drive that uses 5.25-inch cartridges that look a lot like 3.5-inch floppy disks. They happen to contain flexible media that resemble floppies, as well, but the read/write technology is a bit different. Instead of a pair of read/write heads that press down on the magnetic medium from either side, as on a conventional floppy, the medium is actually drawn to the head by the force of the flow of air generated as the disk rotates. A power loss causes the disk to fall away from the head, so a data-damaging head crash is almost impossible. Bernoulli systems are extremely reliable.

Figure 4.3



The Iomega Bernoulli Box II, an extremely convenient open-ended mass storage option.

Yet, each cartridge can store 44Mb of data—as much as an average hard disk. I pay $80 each for my cartridges, which makes them a lot more expensive than tape on a per megabyte basis, but quite a bit less expensive than removable hard disk cartridges.

The performance compares favorably with hard disks, too. I usually don't bother running performance metrics that only serve to make me dissatisfied with the equipment I have. I did, however, compare the Bernoulli Box and my hard disk for a typical file: it took 20 seconds to load from my hard disk and 26 seconds from the Bernoulli cartridge. Not a big enough difference to lose sleep over.

What is important is convenience. Your system treats the Bernoulli Box as just another hard disk. I don't need special backup programs to compress and archive my hard disks when I back them up. With a large disk divided into 44Mb volumes, I can copy the entire hard disk to the appropriate Bernoulli cartridge.

This isn't the cheapest way to go, nor even the fastest, but it certainly is convenient. I don't need to unarchive any files that I want to retrieve from the Bernoulli cartridges; they are stored in their normal format.

I also keep my most frequently used clip art in a separate set of Bernoulli cartridges. Another holds installed versions of various scanner programs and image editing software that I rarely use but might need someday. Another group of cartridges stores gray scale and color scanner image files. You may not be able to store a large number of color images on a 44Mb cartridge, but you'll rarely run into an image that won't fit.

The cartridges also provide a convenient way to transport data from one computer to another. You can even move files between unlike platforms, such as a Mac and a PC, if you have the system installed on both.

Disk Doubler

Disk Doubler and similar programs are tools of interest to scanner users – particularly those who have large quantities of PostScript files. In one sense, such programs are file compression utilities like STUFFIT, which work in a particularly useful way. I want to discuss this category of product here, because they can provide some largely imaginary gains unless you understand how they work.

Most compression utilities generate archive files, which can contain entire collections of compressed files. The archive file can be copied from disk to disk, its contents listed, and files added or deleted as you want. Disk Doubler works in the background, automatically compressing and uncompressing files as you use them.

The process is quite transparent. Disk Doubler slows down access to files a little, but you may not notice the difference. However, it isn't magic; it won't further compress files that are already compressed. In my tests, RIFF files compressed no more than 10%, and some TIFF files no more than 15%. That doesn't mean that scanner owners are left out in the cold. Bit map files that have been converted to PostScript EPS format compress quite well. EPS files are, after all, just ASCII text files with PostScript language descriptions of how to reproduce the images. These will often compress 50 to 75% or more.

So, Disk Doubler can be a useful archiving tool for scanner owners who have large collections of disk-hogging EPS files.

Backing Up Large Images

When backing up scanned images, you may find that some won't fit on a single floppy disk, particularly if you have low density disks 800K disks. How do you copy, back up, or transport a 3Mb image if you are limited to these media—or even if you have a 1.44Mb FDHD drive?

The answer is to use a backup program like Fastback. Such programs will automatically prompt you to insert a new disk when the last one is full. Large files will be divided up among as many disks as required. When you restore them to your hard disk or that of another computer, they'll be reassembled as required.

Summary

Hard disks provide a way of permanently storing scanned images. While 40Mb drives are standard for character-based applications, a scanner user should plan on getting an 80Mb or larger hard disk for serious work.

Generally, hard disk storage gets less expensive on a per megabyte basis as the size of the disk increases. A 40Mb drive may cost $400, and an 80Mb drive with twice the capacity only $500, while a 330Mb drive can be purchased for $1400 or less.

One way to fit more scanned images on your hard disk is to use some sort of image compression scheme. Many image editing packages offer the option of compressing files automatically when they are stored, usually using an industry standard algorithm. You may save as much as 50% of the disk space that would have been required. The only penalty is the increased time required to store and retrieve compressed files.

Image compression is a way of encoding the binary numbers that represent an image so that a smaller number of binary digits is required. When squeezing a program, the compression/decompression process cannot afford to lose a single bit of information. Single bits are rarely so important in image files. Gray scale images rarely change abruptly from one tone to another. There are long stretches of pixels that have the same or a similar value, and gradients are often very predictable. Image encoding schemes can take this predictability into account.

The Joint Photographic Experts Group (JPEG) has developed a compression scheme for continuous tone images that is efficient, and still retains the valuable image information. JPEG uses three different algorithms—one called discrete cosine transformation (DCT), a quantization routine, and a numeric compression method like Huffman encoding.

Your hard disk storage requirements will eventually expand to fill up your available hard disk space. Floppy disks and tape cartridges are popular choices for offline storage of images. Removable hard disks are another solution whose time will never come. Even the most rugged removable hard disk media suffer from rough treatment, and they are the most expensive per megabyte.

For convenience, flexibility, and reasonable cost per megabyte, there is currently only one option: Iomega's Bernoulli Box. This is a SCSI disk drive that uses 44Mb 5.25-inch cartridges that look a lot like 3.5-inch floppy disks.

The performance compares favorably to hard disks. The cartridges also provide a convenient way to transport data from one computer to another.

Disk Doubler is another tool of interest to scanner users—particularly those who have large quantities of PostScript files. In one sense, it is a file compression utility like STUFFIT, but it works in a particularly useful way. It automatically compresses and decompresses files in the background, as you use them. Any file compression program won't further compress a file that has already been compressed by your application. However, file compression programs are very useful with Encapsulated PostScript files and image files that have not already been compressed.

5

The Hand Scanner Revolution

I've always liked hand scanners. Having any scanner is better than having no scanner at all, in my opinion, and I have long recommended hand scanners to anyone who couldn't afford a "real" (flatbed) scanner. After all, you could always use them to scan little bitty pieces of non-critical artwork for position only (FPO) in desktop publishing. They made interesting toys and might even whet your appetite for more upscale, full-featured models.

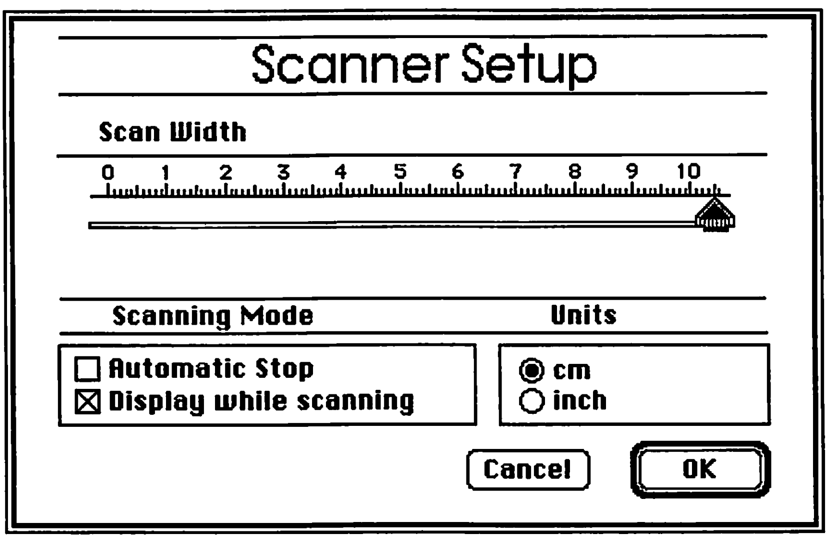
That was then; this is now. Today's hand scanners are starting to make flatbed models look a little like bloated dinosaurs. You can now do just about anything with a hand scanner that you can do with a desktop model, and some tasks can be carried out more quickly and easily. I'm finding it difficult to convince many users even to consider a desktop scanner. If you don't use a scanner all day or every day, why should you have one on your desk? And why should you pay $1000 or more when a $200 hand-held device will do the job. I don't have a good answer. I find I am using a hand-held scanner more and more myself.

The computer I am using to write this chapter is equipped with two scanners. An Epson ES-300C sits a few feet to my right, while a Logitech ScanMan is tucked on top of the Mac II next to the monitor. It plugs into a SCSI interface that is stacked between an external hard disk and a Konnect Drive 2.4.

For some tasks, it's hard to choose between them. I can capture 256-gray level scans with either one. Both take only a second to set up. I can either lift up the cover of the flatbed and put the copy face down as I would with a photocopier, or I can reach over, grab the ScanMan, and slide it across the artwork I want to capture. Both scanners can be operated from a desk accessory, so I don't need to load any special software to use either of them.

Both scanners can be operated from a desk accessory, so I don't need to load any special software to use either of them. Logitech's scanner Desk Accessory is shown in Figure 5.1.

Figure 5.1



The scanning Desk Accessory provided with Logitech hand scanners for the Macintosh.

True, the Epson can capture an 8.5 x 14-inch document in one pass, but the 105mm wide swathe of the ScanMan isn't limited to 4-inch wide strips. It's relatively easy to scan larger images in several passes and then combine them almost automatically.

"Ah," you say, "hand scanners are more difficult to use than flatbeds, because you must be careful to scan at a constant speed, while keeping the scan path perfectly straight. That's too much to ask of shaky, inexperienced hands."

"No problem with the ScanMan," I reply. It has a green LED on top. As long as you're scanning at an acceptable speed, the green light shows. At 100 dpi, you can move about 2 inches per second. At the highest resolution you'll need to slow down to a crawl (about half an inch per second). That pace was actually the only one I had any trouble maintaining, and a little practice solved the problem.

Muscle-Powered Scanning Is Easy

You don't really need to worry about how even your movement is, as long as you don't jerk the scanner, move it backwards at any time, or roll it over an original placed on a bumpy surface. That's because the scanner has a broad roller located just behind the scanning array, which is synchronized with the scanning software. The scanner reads a line only when the roller tells the software that you have moved far enough to bring a new line into view of the sensor. So, a hand movement that accelerates or slows down won't cause the scanner to skip lines or read a line twice. You may even be able to scan a portion of an image, stop, and then restart without causing serious defects in your finished image.

You don't need to press and hold the scanning button on the hand scanner as you move it, either. Just a single press starts the scan, and you can hold the scanner comfortably without squeezing for the rest of the pass. If your scan is a bit crooked, you can correct it easily with the anti-skewing features of many software packages.

Not convinced? My hand scanner has three controls, which allow me to configure it without bothering with a bunch of dialog boxes and menus. I can select 256, 64, or 16 gray levels, as well as a binary, 1-bit, scan with one slide switch. A rotary dial adjusts contrast. Another slide switch changes from 100 dpi to 200, 300, or 400 dpi. What's that? Your flatbed scanner doesn't have a 400 dpi setting? Don't fret. As we'll see later, really high resolutions aren't needed very frequently. They are handy to have available, though.

If the ScanMan isn't wide enough for you, or you don't like taking the time to stitch images together, you can buy a hand scanner with an 8-inch head. In addition, a growing number of full color hand scanners are reaching the market, and optical character recognition (OCR) with hand scanners is becoming practical for the first time. Flatbed scanners still have some advantages, but you no longer have to purchase a hand scanner just as a way of saving money. Many users have found that a $200 to $400 hand scanner will do everything they need to do, efficiently. Why pay twice as much?

The software available for hand scanners is every bit as sophisticated as that used with flatbed, sheetfed, and overhead scanners.

Tips for Hand Scanning

Although I presented some tips for getting the best scans with a hand scanner in my last scanner book, I've learned a few things since then. Some new ideas came from Logitech, which pointed out factors I hadn't been aware of. The examples in the list below illustrate some of the finer points you probably weren't aware of.

- When using a hand scanner, don't move your mouse or do any printing at the same time. These might not appear to be likely events, but it is easy to bump your mouse if it shares the same desktop as your hand scanner (mine doesn't).

  No, it isn't the vibration of these two activities that disturbs the scan; it's the extra call that your computer makes to the peripheral that can interrupt the scan. Your finished image may have corrupted data. That's no great problem: you can just rescan.

- As I mentioned earlier, you may need to slow down your scanning speed drastically at higher scanning resolutions, particularly 300 to 400 dpi. Earlier hand scanners often interpolated higher resolution data from lower density scans—typically no higher than 200 dpi. Today's hand scanners really can scan at a full 400 dpi and require you to slow down, so they can capture and pass along the information they receive.

  The best speed is determined partly by the type of image you are scanning. A binary, black-and-white image doesn't contain as much information and can be scanned more quickly. A full 256-gray level or color scan produces much more data, calling for a reduced scanning speed.

- You probably don't need to grip your hand scanner in a choke hold. The rollers built into the latest scanners provide a sort of self-guiding mechanism that make it easy to scan smoothly. If you press down too hard or hold the scanner too tightly, you can generate conflicting forces that cause the scanner to move offtrack or to jump when it comes to an edge or a slight defect in the original. My own hand scanning technique typically involves more push than drag.

The Future of Hand Scanners

Someone may have said that the future is not only stranger than we imagine—it is stranger than we *can* imagine. I'm sure that Theodore Sturgeon or another futurist said something along those lines. The point is that we can scarcely visualize what direction computer technology will take in the next few years.

The $2000 to $3000 desktop scanners of a few years ago that could capture only 16 levels of gray were seen as specialized devices with limited uses only a few years ago. Today, you can scan 256 gray levels for $400 or less and color for $600.

So, the hand scanner revolution may take us in some directions we didn't expect. For one thing, a hand scanner may become a standard part of a Macintosh configuration, just as a mouse is today. Make a scanner small enough, cheap enough, and powerful enough, and every user will have and use one.

For example, imagine a scanner and mouse combined. In its default mode, your mouse/scanner would be small enough to fit in your hand, like current mice, but wouldn't be linked to your computer by a cord. Instead, more reliable infrared signals than are available today would be used. The mouse/scanner could send information about mouse movement direction and speed, just as a mouse does today. However, the scanner component could be used to provide precise positional information when used with a special mouse pad. That is, you could move the mouse pointer to a specific spot on the pad to trigger a command or to perform tracing functions. In that respect, a mouse/scanner would function much like a digitizing pad.

The average mouse is about 2 inches wide—hardly a reasonable swathe for scanning pictures or text. Yet, it might be useful for grabbing bits of images or information from a form, making the mouse/scanner a new sort of input device for some types of software.

At the press of a button or two on the mouse, a wider sensor would slide out, extending perhaps an extra inch on each side of the mouse/scanner and allowing the same roughly 4-inch wide pass provided by the typical hand scanner. Improved stitching algorithms would let you slide the mouse/scanner over an image several times with little regard for the amount of overlap, skew, or scanning speed.

The software would then assemble a complete image automatically, discarding the overlapping sections, reorienting skewed images, and fixing discontinuities where you accidentally jiggled the mouse during a pass.

You might even be able to select the function—OCR, line art scan, gray scale scan, or color scan—by pressing a combination of mouse buttons. A two-button mouse can send eight different commands just by clicking or double-clicking the right, left, or both buttons in the proper sequence. You could get a couple more functions by assigning double-click/drag combinations with special meanings. A three-button mouse offers even more possible combinations. I haven't even mentioned the possibilities of Shift-click, Ctrl-click, and Alt-click mouse/keyboard sequences. Such schemes would be preferable to those "power" mouse abominations I've seen with 16 or 20 different buttons on them.

Hand scanners of the future could let you choose drop-out color, provide even higher resolutions, and function as input devices that would replace today's fax machines and personal photocopiers. If these predictions seem strange to you, remember that the hand scanner of 1999 may be a lot stranger yet.

Summary

Hand scanners once had limited functions and fit best into certain narrow applications. Today's models can do just about anything a desktop model can do and, in many cases, do it more quickly and easily.

We're also seeing specialized hand scanners for special applications, such as OCR, and a new breed of low cost color hand scanners.

Hand scanners work best when you follow some simple guidelines:

- When using a hand scanner, don't move your mouse or do any printing at the same time. The extra call your computer makes to the peripheral can interrupt the scan.
- You may need to slow down your scanning speed drastically at higher scanning resolutions, particularly 300 to 400 dpi.
- Don't hold the scanner too tightly or press down hard while you scan.

A hand scanner may become a standard part of a Macintosh configuration, perhaps combined with a mouse. At the press of a button or two on the mouse, a wider sensor would slide out, extending perhaps an extra inch on each side of the mouse/scanner and allowing the same roughly 4-inch wide pass provided by the typical hand scanner. Improved stitching algorithms would let you slide the mouse/scanner over an image several times with little regard for the amount of overlap, skew, or scanning speed.

Hand scanners of the future could let you choose drop-out color, provide even higher resolutions, and function as input devices that would replace today's fax machines and personal photocopiers.

6

What Scanner Vendors Don't Tell You About Installation

If you've always used Macintoshes and nothing else, you may not realize just how lucky you are. One of the most vexing things about the "other" desktop platform is the difficulty you encounter in doing relatively simple things, like installing hardware or software.

Following the vendor's instructions is no guarantee of success. That's particularly true when installing a scanner in an IBM compatible system. You have to worry about port assignments, interrupts, types and amounts of memory available, various graphics standards, and other factors. Just installing software can take half an hour, and a couple dozen questions.

Mac installations, in sharp contrast, are a lot simpler. Scanners most often use the SCSI interface built into every Macintosh since the Plus. All you need to do is select a SCSI assignment, and you can often do that by setting a few DIP switches on the scanner. Perhaps 90 percent of the scanner software I use is installed by dragging a few folders from floppy disks onto one of my hard disks. In most cases, there's no need to open the Macintosh or scanner. You can often install a scanner and its software in five minutes or less.

Yet, there are a few things you need to know. This section tells you some of the things that you may need to know to get your scanner working but which the scanner vendor won't bother to tell you.

I don't want to paint too bleak a picture; most scanner installations go off without a hitch. You needn't worry that getting a scanner to work is so hopelessly complicated that you have no chance at all of completing the job. This chapter is intended only for those who do run into problems or who must install scanners in a variety of equipment with heterogeneous configurations.

In most cases, being armed with information is your best defense, but getting that information can be a struggle. Have you ever had a conversation like this with the technical support staff of your favorite vendor? The two examples below happened to me within two weeks:

Conversation #1:

DDB: I've called all around the country and none of your dealers has an interface for your original scanner in stock. They say they can't order it from you.

Tech: Oh, you can use the interface for our latest scanner. It is more compact, but other than that it's just the same. Everything is identical.

DDB: I called last week, and one of your people said I needed the original interface.

Tech: That's not right. Just use the new one.

DDB: I can use your latest scanner drivers, too?

Tech: No, you'd better stick with the older drivers for the original scanner. We've changed them, but they should work with the newer interface.

I ended up buying the new interface, using the old drivers, and finding that everything worked just fine except that the scanner acted funny sometimes. It took me a while to figure out that the initialization routines were different enough that I now needed to wait to turn the scanner on until after my software was loaded. I hadn't had to do that before.

This next section will tell you some of the things you need to know at 1 a.m. on a Sunday morning when all the tech support lines are shut down and you just *have* to get your scanner working.

Scuzzy or Sexy?

The folks who helped originate the interface would like you to think that SCSI stands for sexy, but all the world refers to this particular set of standards as scuzzy. The acronym actually stands for small computer systems interface, and SCSI, which once had a tarnished reputation because it was one of the most non-standard of computer standards is now back in favor.

It actually never left favor with Apple Computer, which has included the SCSI interface in all its Macs since the Plus. While SCSI is most often used for hard disk drives in the IBM world, we Mac users hang everything from scanners to external high density floppy disk drives on our SCSI ports. It is because the Mac's SCSI interface has so many applications that you need to know a little about it, and how it works.

SCSI is what is called a *system-level interface*, which conveys information to the CPU in logical terms. That's in contrast to the so-called *device level interface*, which sends information along dedicated wires or lines. Because devices on a SCSI interface each don't require their own set of dedicated lines, they are able to share a single bus in parallel fashion. More intelligence is required in the device itself, as the device must decode requests from the computer and decide that the signal is meant for it and not one of the other devices on the connection.

Each SCSI device must have circuitry on-board that receives requests for information from the Mac, and intelligently handles (in the case of a hard disk) finding the data, retrieving it, decoding it, and passing it along to the computer on the data lines common to all the SCSI devices in the chain.

Because the computer doesn't have to be concerned with the nuts and bolts of operating the peripheral, a SCSI device can be a hard disk, a tape drive, an optical disk drive, some printers, or a scanner. On faster Macs, the SCSI bus can transfer information as quickly as 1.4Mb a second. That's ideal for the huge amounts of information that can be produced during a color or gray scale scan.

Up to eight devices can be attached to a SCSI bus, but one of them is always your Macintosh, so that leaves you seven connections for your peripherals. SCSI devices aren't attached to a single cable with multiple connections on it. Instead, external devices are daisy-chained together. That is, you plug a cable into your computer, and then into the first device on the chain. To add a second device, plug an additional cable into a connector on the first peripheral, and then into your new device.

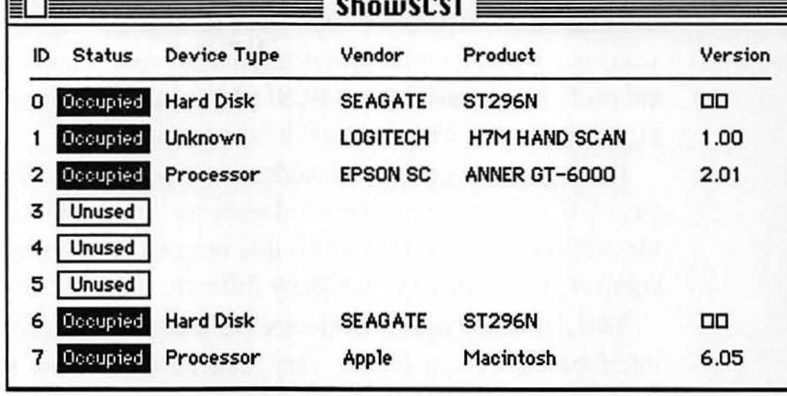
Each device in the chain has an address number, from 0 to 7; the higher the number, the higher the priority that device has. When the Mac has information coming in from more than one SCSI device at once, the one with the higher address priority can be handled first. The Mac itself always has the SCSI address of 7, and therefore is given the highest priority in the chain.

The SCSI address assignment does not have to relate to the order of the devices on the chain, nor do you need to use them up in any particular order. An external hard disk that is closest to the computer may be Device 5, while a scanner that follows it on the chain may be Device 2 or 6. It doesn't matter. What *does* matter is how many and which devices are terminated, but I'll cover that shortly.

There are a number of utility programs that that can "read" the SCSI bus and tell you what devices are present. I use ShowSCSI, which is provided as part of the Fastback II package. It produces a display like that shown in Figure 6.1, produced by one of my Macs, an original Mac II. The program can interrogate the devices to retrieve information about them. The manufacturer can hard code the model name and number of the device to make this data available, if so desired. There are other utilities, either free or available for a nominal shareware fee, that will also tell you about your SCSI devices. These include SCSI Identifier (a simple application), SCSI Tools (a Control Panel device), and, the most sophisticated, SCSI Evaluator, which will also test your drive and provide a report. You can usually find these on BBSes or other places that deal in shareware.

Using ShowSCSI, for example, in Figure 6.1, you'll see that Device 7 is, as you'd expect a Macintosh II, of the device type *processor*. Even the System software version is available. Macintoshes after the Mac II

Figure 6.1



ShowSCSI

ID	Status	Device Type	Vendor	Product	Version
0	Occupied	Hard Disk	SEAGATE	ST296N	□□
1	Occupied	Unknown	LOGITECH	H7M HAND SCAN	1.00
2	Occupied	Processor	EPSON SC	ANNER GT-6000	2.01
3	Unused				
4	Unused				
5	Unused				
6	Occupied	Hard Disk	SEAGATE	ST296N	□□
7	Occupied	Processor	Apple	Macintosh	6.05

The Show SCSI utility program, available from BBS's and other sources such as CompuServe.

(this includes the SE and members of the "new" generation like the LC) have two SCSI ports. One is a 50-pin ribbon connector inside the case which can be plugged into an internal hard disk, in my system an 84Mb Seagate ST296N, and designated as Device 6.

Macs from the Mac Plus on have an external SCSI connection, a female DB-25 connector on the rear panel of the computer. I have three devices daisy-chained in this particular system, which is used primarily for scanning; an Epson flat-bed scanner as Device 2, a LogiTech ScanMan as Device 1, and a second Seagate disk drive as Device 0. (Incidentally, both drives were removed from an IBM computer, moved over to the Macintosh – one of them to a case with a power supply and connectors, of course – and then reformatted using the Mac version of OnTrack Disk Manager. I made no other modifications other than to set the address number of each. It's easy to see why Macintosh hard disks no longer cost a $200 or more premium over their IBM PC counterparts; there isn't really any difference between them.)

Your scanner will probably use the Mac's external SCSI connector, and should be furnished with a cable that has a male DB-25 connector on one end, and a 50-pin connector on the other. If it is your only external SCSI device, simply plug in the cables to the port on your Mac with the diamond-shaped icon over it, connect the other end to the scanner, turn on your computer, and install your scanner software. You are *probably* in business. A SCSI address that should not conflict will probably already have been set by the manufacturer.

If not, you can change the address of one or more devices using the steps I'll describe below. If you already have one or more SCSI devices, the odds that you'll need to do this increase, and you'll also need to connect your scanner in a slightly different way.

To daisy-chain one SCSI device from another, you need a peripheral interface cable, which may vary, depending on how your particular device is configured. Fortunately, the connectors are standard, even if the way they are used with various SCSI devices isn't, so you should have little trouble obtaining the cables you need.

Some devices will have 50-pin connectors on the back for an incoming connection, and a second for outgoing connections. Others will have a 50-pin connector coming in, and a DB-25 going out. Either setup will work in a daisy chain, but you obviously need different cables for each configuration. A few devices won't have any removable incoming cable at all, usually to save the space required for a connector for detachable cables.

For example, my Logitech interface box has a cable with a male DB-25 connector permanently wired to it, for incoming signals. It features a female DB-25 connector for daisy chaining to the rest of the bus. My Epson scanner, on the other hand, has a 50-pin connector coming in, and a female DB-25 going out. The external hard disk has 50-pin connectors coming and going.

This may sound confusing, but it really isn't. The Logitech interface—because it has a permanently-wired incoming connector—has to be plugged into a DB-25 connector, either the one on the back of the Mac, or the one on the Epson scanner. The Epson scanner and the external hard disk can be connected any way you like; you just need cables that match the "source" and "destination" for your SCSI connections.

Some mail-order firms offer a good selection of cables you can use. I like MacWarehouse because their catalog lists all the combinations on a separate page in a table that is easy to interpret. I also like being able to order an in-stock item until late in the evening and still receive it by 10:30 a.m. the next morning for only a $3 Federal Express charge. You can get any of the cables I discussed, plus SCSI cable extenders that let you move a peripheral a bit farther from your computer, and external terminators that can be plugged into the "out" connection of one of your devices, if necessary.

The Terminator

Arnold Schwartzenegger coined his most famous tag line in his enjoyable film *The Terminator.* When he promised "I'll be back!" the other characters had no inkling of how much trouble that would cause. You don't want signals to come back in a SCSI chain, either, unless you want trouble of a very different kind. On the SCSI bus, a terminator is a device which is used to absorb SCSI signals at the end of the chain to keep them from bouncing back to produce electrical noise that can lead to problems like disk errors.

Generally, your SCSI bus should have two terminators, one at the beginning and one at the end of the chain. If your computer has an internal hard disk, it should have a terminator. If your scanner is the only external device on the SCSI bus, it should have a terminator, too.

What if you have a Mac Plus with no internal hard disk, but have both an external hard disk and a scanner connected? In that case, both external devices should have a terminator of some sort.

Things get only a little more complicated when you connect have an internal hard disk, and two or more external SCSI devices—say an external disk and a scanner. In that case, the device (or devices) in the middle doesn't need a terminator, but the one at the end of the chain does.

So far, things aren't really very confusing. The actual fly in the ointment comes from the fact that devices can have terminators built in, or may require you to use an external terminator. You may not know which type should be used without consulting the device's manual or, worse, opening the device itself.

On devices like hard disks, internal terminators often look like a set of resistors encased in a glob of plastic, with eight small pins sticking out. To remove the terminator, you simply pull the package out of its socket. The most thoughtful vendors put a panel that you can open on the case immediately above the terminator. Instead of taking the whole device apart, you can open the panel and remove the resistor pack. The *very* nicest vendors put a transparent plastic window in the panel so you can tell at a glance whether an internal terminator is installed, without removing the panel itself.

If you want a device to be terminated, but it doesn't provide for internal termination, you need to use an external terminating connector, which is placed between the incoming SCSI cable and the connector on the device. Some external terminators just plug into the outgoing SCSI connector. Either kind will work just fine.

Although the last device on the chain must be terminated, which device that is may be determined by how you use your system. Both the first and last devices in the SCSI chain must be powered on when the Macintosh is started up for the SCSI bus to function properly. The devices in the middle of the chain can be switched off until they are needed.

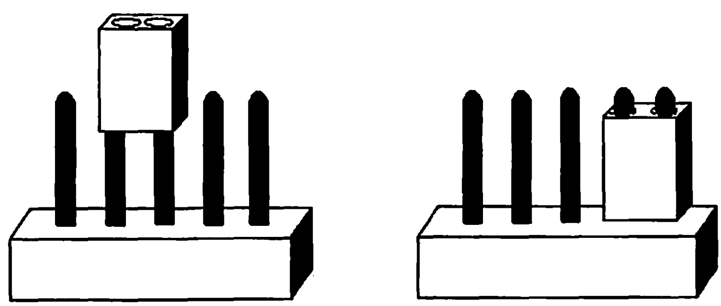
For that reason, my internal and external hard disks are the first and last peripherals in the chain, respectively, because I always want them available when I am using the Mac. Both scanners are placed in the middle; I can power them up and use either or both of them as I need to.

Setting Device Numbers

Setting a device number for your scanner that doesn't conflict isn't especially difficult, even though various systems are used by various devices. Some devices, such as hard disks, use a jumper to specify a SCSI address. A typical jumper is shown in Figure 6.2. In a hard disk, the SCSI jumper is often located on the back of the disk, near the cable connector.

Other devices, including some external hard disks, use a more convenient scheme, either a rotary switch on the back of the unit, which can be set using a small screwdriver, or a pushbutton device that cycles through each address as you press it one or more times. In both cases, it isn't necessary to open the device to set the SCSI address. You might also need to set a DIP switch or thumb switch to specify your address. Regardless of the scheme used, you should use a utility like Show SCSI to double-check what you've done, or carefully check visually, to avoid address conflicts.

Figure 6.2



A typical jumper, like you might find inside a SCSI device.

Summary

Scanners don't always install as easily as the pretty ads in the magazines lead you to believe. This chapter explained some of the things that scanner vendors don't always tell you, particularly about the Small Computer Systems Interface (SCSI).

Apple has included the SCSI interface in all its Macs since the Plus, and it is used for everything from scanners to external high density floppy disk drives on our SCSI ports.

SCSI is what is called a *system-level interface*, which conveys information to the CPU in logical terms. Because devices on a SCSI interface each don't require their own set of dedicated lines, they are able to share a single bus in parallel fashion. More intelligence is required in the device itself, so it can decide that a signal is meant for it and not one of the other devices on the connection.

Up to eight devices can be attached to a SCSI bus, but one of them is always your Macintosh. Each device in the chain has an address number, from 0 to 7; the higher the number, the higher the priority that device has. The SCSI address assignment does not have to relate to the order of the devices on the chain, nor do you need to use them up in any particular order.

Macintoshes after the Mac II (this includes the SE and members of the "new" generation like the LC) have two SCSI ports, one internal and one external. Your scanner will probably use the Mac's external SCSI connector, and should be furnished with a cable that has a male DB-25 connector on one end, and a 50-pin connector on the other.

To daisy-chain one SCSI device from another, you need a peripheral interface cable, which may vary, depending on how your particular device is configured.

On the SCSI bus, a terminator is a device which is used to absorb SCSI signals at the end of the chain to keep them from bouncing back to produce electrical noise that can lead to problems like disk errors.

Generally, your SCSI bus should have two terminators, one at the beginning and one at the end of the chain.Terminators may be internal or external devices.

Both the first and last devices in the SCSI chain must be powered on when the Macintosh is started up for the SCSI bus to function properly. The devices in the middle of the chain can be switched off until they are needed.

Some devices use a jumper to specify a SCSI address. Other devices use a more convenient scheme, either a rotary switch on the back of the unit, which can be set using a small screwdriver, or a pushbutton device that cycles through each address.

7

Compiling Your Own Clip Art and Photo Library

If you went to the dictionary to look up the term "Renaissance Man," you'd see my friend Merrill. He would probably be there ahead of you, looking for some line illustrations to add to his clip art library. He's as passionate about clip art as most of us are about our spouses, and he spends more money on it than most of us spend on our spouses. Of course, clip art is more than a consuming interest for him – he uses it to spice up his desktop-published products.

I always pay attention to what Merrill is interested in. With his brother, he invented the concept of video game rooms in shopping malls – in the late 1960's, before there were video games and only a year or two after the first shopping malls began to appear around the country. Later, he founded an on-line computer service a few years before General Electric and Sears decided to get into the business. I made my first scan on a flatbed scanner at his house and then decided to get one for myself. The only time Merrill isn't thinking two years ahead of the rest of the world is when he is thinking ten years ahead.

By that token, clip art is bound to be a growing industry in the coming decade, if only because of Merrill's financial contributions to the various vendors. Scanners should help fuel that growth, since they provide a convenient way to assemble a vast clip art library that can add visual interest to your desktop publications.

What Is Clip Art?

Clip art illustrations are generic drawings, usually prepared by professional artists. Traditionally, clip art has been published in hardcopy form for use by other graphics professionals. Clip books are printed on glossy paper on only one side of the page, so the images can be cut out and pasted directly on a finished mechanical. Frequently, each illustration is supplied in several sizes, so no additional reduction or enlargement steps are required in many cases. The mechanical—text and art both—is then photographed to produce the page negatives that expose the printing plates.

Clip art is most frequently black-and-white line art, but more advanced renderings may also be included in some collections. Figure 7.1 shows a sample commercial clip art image.

The photographic equivalent of clip art is the stock photo. A stock photo is a generic photograph that can be scanned and used in a publication much like clip art, although the additional step of halftoning is required to make an image that can be reproduced on a printing press or other output device. While I'll refer only to clip art in most of this chapter, most of the advice you'll find here also applies to stock photographs.

Few desktop publishers amass stock photograph collections, because the rights to professionally produced photos are generally purchased on a one-time-use basis. You peruse a catalog of available images and then purchase the rights to the photo you want from the stockhouse vendor as the photo is required.

Why Use Clip Art?

Why do you need clip art for your publications? Because few of us are accomplished artists, and there are very few publications that couldn't benefit from some artwork to illustrate a point or two. Such illustrations are almost always better than nothing from a visual standpoint, and they are often right on the money.

Figure 7.1

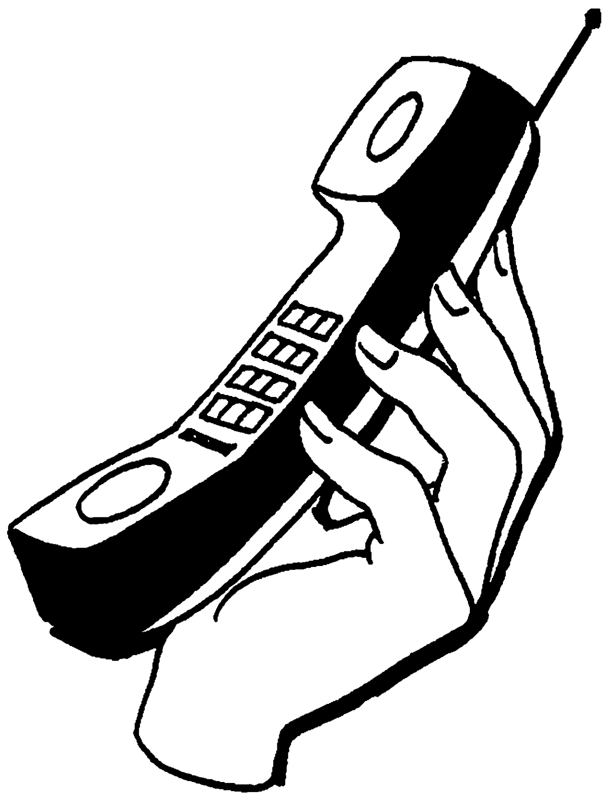
An example of commercial clipart, scanned from Clipper. (This copyrighted material has been provided by Dynamic Graphics, Inc., 6000 N. Forest Park Drive, Peoria, IL 61614 and may not be reproduced in any way without the express written permission of Dynamic Graphics, Inc.)

For example, if you're doing a piece on desktop publishing, you may want an illustration or two that shows how to position artwork on a flatbed scanner. If you can't prepare the drawings yourself, a clip art illustration of an appropriate scanner will at least provide some visual interest in the section. It breaks up dreary columns of unaccompanied text and helps readers visualize what you are talking about in your word pictures.

However, you may be able to do better. If a scanner is too complex for you to draw from scratch, why not try to take the basic clip art drawing and modify it? Perhaps you can sketch in a few lines that show correct positioning. As long as you can mimic the style of the original drawing so the additions don't stick out, you needn't have professional-level skills. Clip art can help you produce custom-looking artwork by getting you 80 to 90% of the way there. Your own more limited abilities can take you the rest of the way. Figure 7.2 shows an enhanced clip art image.

Most of the time, you will be able to find a piece of clip art that exactly fits your needs, with no changes required. The larger your library of

Figure 7.2



This commercial clip art was first scanned, then enhanced with a paint program. I needed a cordless telephone, but couldn't find one anywhere. So, I added an antenna and removed the cord from this example.

available images, the more likely you are to find a drawing that suits a specific article or publication.

Clip Art Formats

An increasing amount of clip art is available already in digitized form on disks in bit-mapped or PostScript format for Macintosh computers. You can even purchase CD ROMs containing hundreds of clip art images. If you're interested in this sort of art, you'll find dozens of ads for digitized commercial clip art in the back of any magazine with a desktop publishing audience, such as Personal Publishing and Publish.

These vendors provide professional artwork that can be used by those who don't have access to a scanner. Very little of that art actually originates on a computer. Most is created in the traditional way using pen and ink, or other media, and then scanned. Artists prefer to draw that way, and the system has the added advantage of allowing the originals to be photographed and printed as hardcopy clip art for cut-and-paste publishers. (In fact, it's likely that the art was created for that purpose and is being provided in electronic form to take advantage of a new, growing market for clip art.)

However, since you have a scanner, you don't need to purchase clip art in electronic form. Instead, you can collect hardcopy images and scan them as required. The original images can come from printed clip art collections that you purchase or from other printed material you accumulate. See the separate section on copyright considerations for a discussion of allowable uses for this type of art.

Should you scan in the clip art and keep files available, or should you just file the hard copies for later use? There are advantages to both approaches.

Advantages of Scanning on Demand

- **You won't scan images that you don't use.** My friend Merrill probably has tens of thousands of images. Most of them he paid for, since he uses them in a business operation and can justify the cost. He probably will never use a tenth of a percent of them. Like Imelda Marcos's shoe collection, it gives him peace of mind just to know that if he ever does need an image of Benjamin Franklin, he has five or six from which to choose.

If you want the maximum size library with the minimum expenditure of time and money, you'll want to collect hardcopy images. You can retain many more images than you need at little additional cost.

- **You gain flexibility in resolution, sizing, and scaling.** If you scan an image just before you use it, you can scan it at the exact size and resolution that best suits your application. You may need only a portion of an image and want to scan it at a relatively high resolution to preserve detail. A generic, "stock" scan may not be at the highest resolution available, to save on disk space, and you can't work with detail that wasn't captured in the first place.

  This also applies to photography. If you happen to have permanent rights to use a set of photographs, you probably won't want to scan them until they're actually needed. That way, you can perform some of the cropping as you scan and exercise greater control over the range of tones and image contrast.

- **You save a lot of disk space.** Digitized images occupy a great deal of hard disk real estate. Each scanned image requires a minimum of 2K and, as a practical matter, amounts to no less than 8K per image (a tiny 200 x 40 pixel logo, for example). More commonly, you'll find bit-mapped scans occupying 50 to 100K or more. Storage costs for images that you may never use can mount quickly.

  One of my computers has a 330Mb hard disk, and I don't waste a lot of space on clip art. I also own a couple dozen Bernoulli cartridges that hold 44Mb each, but my entire clip art library doesn't reside on that removable storage medium either. My clip art averages about 100K per image, and I pay $80 for each Bernoulli cartridge, so I have to be willing to pay 18 cents for every single image I want to retain in magnetic form. Using 1.44Mb floppy disks would cut the cost in half while making storage of a large collection of images much more cumbersome. My

Konnect Drive 2.4 allows formatting these disks to 2.4Mb, but even that extra capacity doesn't reduce the cost enough.

So, I have set aside a file drawer for storing hard copies of images I might want to use as clip art. Some folders hold commercial clip art booklets and pages torn from them. Others hold different pieces of art I may want to recycle. This archiving method uses space less efficiently than storing images on compact magnetic media, but it costs me almost nothing.

- **Browsing is easier.** A well-organized hardcopy system can be accessed quickly and conveniently. You can rifle through folders, glancing at hundreds of images in the time it takes to load and view only a few bit-mapped files. As the use of graphics increases, the need for better ways to manage and view image files becomes crucial. But until someone comes up with the solution, hard copies are still the best way to compare and select clip art.

Advantages of Scanning in Advance

- **Quicker access to images.** Certainly it takes only a few minutes to find a hard copy and scan it. But what if you don't have a few minutes? Having the clip art you want already scanned and available on a disk can save valuable minutes in critical situations.
- **Ability to re-use an image.** If you plan to re-use an image repeatedly, you're better off storing it in electronic form. You'll save a lot of time not having to rescan it for each use.
- **Filing flexibility.** While hard copies can be stored in folders, you must decide ahead of time what category best suits your image. Moving images around and revamping classifications can be very time-consuming. A given image can be stored in only one place.

  Electronic clip art, in contrast, can be moved around, duplicated, renamed, and reclassified in dozens of ways. System 7's alias featur allows you to "store" files in several different places on your disk without using up a lot of storage space. If you want to store separate copies of a favorite image in several different categories, you can do that.

- **Ability to retain modified images.** If you plan to change an image significantly before you use it, storing the hard copy only delays making these modifications. In some cases, you may postpone the work until you don't have the time to do it. Scan the image now, make the changes, and then store the electronic copy for later use.

Obtaining Clip Art

You can scan anything that can be accepted by your scanner, so, taking copyright restrictions into account, anything printed is potential fodder for your library. This section looks at some of the most fruitful avenues you can explore.

- **Commercial clip art.** This is your best source for generic, professionally done illustrations that you can use without too many restrictions. One thing you can't do is scan in all the clip art in a book and then market it as your own clip art library. You can, however, drop these illustrations into your newsletters, reports, advertisements, catalogs, and other publications.

  You can find clip art books at shops catering to graphic artists and at some newsstands and bookstores. One excellent series I can recommend is published by Dover Publications (31 East 2nd Street, Mineola, NY 11501). The same clip art is published in Canada by General Publishing Company (30 Lesmill Road, Don Mills, Toronto, ON).

  These 64-page, 8.5 x 11-inch books feature more than 200 black-and-white illustrations each. A sample is shown in Figure 7.3. The pages are printed on glossy paper on one side only, with several sizes of each illustration. Scan the largest one for the best resolution. Categories available include office and business, baby and infant care, medical and health services, school and education, nautical and seashore, children, weddings, men's and women's heads, hands, outdoor recreation, sports, and performing arts.

  Another good source of printed clip art is Dynamic Graphics (6000 North Forest Park Drive, Peoria, IL 61614), which offers Clipper, one of the world's leading general purpose clip art services. This 8-page,

Figure 7.3

An example of clip art from Dover, a leading vendor of business artwork that is priced right.

12.5 x 19-inch monthly publication offers 70 individual illustrations, in two or more sizes, along with examples of how to use them. At $32.50 each month, it's a bargain if you use clip art a lot but may be too pricey for the casual user. You do get pictorial indexes, a companion magazine that shows how to use clip art, and durable binders for the art.

Dynamic Graphics also offers Med Art Gallery, a quarterly publication that offers medical and health care illustrations for $180 a year. If you're involved in advertising or promotion, you might also like their Print Media Service, which includes graphics suitable for seasonal events, restaurant and fashion ads, and other applications. The company also offers themed clip art packages in most of the same categories covered by the Dover publications. Figure 7.4 shows a sample graphic from this company.

As you can see, these services cover many of the situations you'd want illustrated by clip art, and it's convenient to have all the images in a given category grouped together. Because you'll be scanning images rather than clipping them out of the book, you'll never use up one of these clip books.

Figure 7.4

An example of clip art from Dynamic Graphics, of Peoria, IL.(This copyrighted material has been provided by Dynamic Graphics, Inc., 6000 N. Forest Park Drive, Peoria, IL 61614 and may not be reproduced in any way without the express written permission of Dynamic Graphics, Inc.)

- **Expired copyright material.** Another source of copyright-free clip art is books published more than 75 years ago. Anything printed before 1916 can safely be scanned and re-used with few restrictions. But you can't, for example, use the registered logo of a company even if you happen to find it in a very old book.

  I regularly haunt used book stores – primarily because I want books to read and secondarily because I collect travel guides to Spain published before 1900. My third reason for visiting these stores is to find books with steel engravings, woodcuts, and other illustrations that can be used in desktop publishing.

  On the one hand, very old clip art probably looks dated. Illustrations of Victorian gentlemen and ladies playing croquet would look out of place in any sort of high tech oriented publication – unless you want to cultivate an old-timey image. These images would be quite suitable for, say, a home-style recipe book or anything dealing with nostalgia. The key is to use antique clip art in an appropriate manner and not to mix it with a more contemporary look in the same publication, chapter, or article. Figure 7.5 shows an old image I collected on one of my forays.

  You'll find that even subjects that you might consider timeless will have a look that dates them. Artists' styles have changed enough that modern renditions of Greek mythological figures or people from history who lived before the advent of photography look different. Even medical illustrations have a special look, despite the dearth of organs discovered in the last century.

  The best old books to look through are dictionaries and encyclopedias. They have many small illustrations, and you can look for the topic you want alphabetically! Text books that cover the subject you are interested in also make good sources. Don't overlook old college yearbooks for school and sports drawings with a period look. I've found these for as little as 10 cents. Most of the other old books I use cost no more than two or three dollars. These editions are often damaged and of no use even as reading copies, so you don't have to feel guilty about dismantling them.

  Indeed, some books are worth more in pieces than they are whole. Another of my friends buys broken hand-illuminated medieval books

Figure 7.5

An illustration from *Longfellow's Complete Poems,* originally published between 1863 and 1902

and sells each of the individual pages as a framable work of art for more than he paid for the whole book. I've managed to borrow a few to scan as clip art; I know for certain that the 16th Century copyright has long since expired.

- **Public domain work.** Strictly speaking, works on which the copyright has expired pass into the public domain. However, some illustrations are created specifically for unrestricted use and are never copyrighted at all

  Government publications are a prime example. You can often use publications prepared at public expense without any restrictions. You may even reprint them and sell them for a profit if you wish. Several firms shamelessly reprint official Federal Income Tax publications and sell them for several times what the IRS charges. Others annotate them or translate IRS-ese into English, thereby providing a valuable service.

You can use illustrations from government publications in your own work if you wish. The Superintendent of Documents in Washington, DC (Government Printing Office, North Capitol and H Streets, NW, Washington, DC 20401) can provide you with a list of available publications. Many are free, while others cost a few dollars.

The government also has many sources of photographs and other non-line art that you can obtain at low cost and use in your publications. The resources are too extensive to list here. Try the individual agencies that supervise your areas of interest. For example, the U.S. Forestry Service has an extensive library of aerial photographs. Satellite and space photographs can be obtained from several agencies. Figure 7.6 is an example of government artwork.

- **Copyrighted works.** Often, you may be able to scan and use copyrighted material legally simply by asking the permission of the copyright holder, who may ask you to include a credit line. Rights to materials published in magazines may belong to the publication, the artist, or the photographer. Start with the magazine, unless the artwork has a credit line you can easily track down. Only line art can be usefully scanned from publications. If you want to reuse a photograph, you'll have to get the original to scan.

- **Reference libraries.** I've put the reference departments of public libraries, universities, and businesses as a separate category because they are often overlooked as sources for clip art. You'll find a great deal more than books in these libraries. They also hold brochures, photographs, drawings, and other materials.

  Don't confine yourself to the obvious areas. Each university department may have a library of it's own. The School of Architecture may have more drawings of buildings than the university's central library. Some of them may be student work that you can use with permission at no cost.

Figure 7.6

The government is an excellent source of military and related artwork, but you'll also find illustrations suitable for more general business applications in government publications.

Businesses typically have libraries of their own, often specializing in a particular industry. If I wanted some baby pictures or similar material that I couldn't find elsewhere, I'd probably check with Gerber before giving up.

You say libraries won't let you take reference material home to scan? Try photocopying the material you need and then touching it up with an image editing program. You may be able to restore any quality lost in the photocopying process.

Or, take your scanner to the library. If you have a hand scanner and one of the transportable Macintoshes,that's not a farfetched suggestion.

You may be able to set up such a scanning configuration in a few square feet of desktop and scan images without disrupting the library too much.

Copyright and Legal Considerations

I may be the first one to have noticed it, but the scanner has rocked the foundations of our legal system. Things you used to be able to take for granted no longer can be relied on. Chances are, if you saw a photograph in a reputable publication a few years ago, you'd believe what you saw with your eyes. Supermarket tabloids aside, most publications had neither the resources nor the interest to do major overhauls on the images they published.

Today, anyone with a scanner and a gray scale editing package can produce a reasonably good picture of, say, Elvis chatting with Dan Quayle while Jimmy Hoffa peers over his shoulder. A scanner and a color printer can produce $20 bills good enough to earn a free vacation at government expense in a penal institution near you. If you need some nice receipts for your imaginary tax return, you can quickly scan in the logos of your favorite suppliers and use your desktop publishing software to print them up.

I still laugh at the teenagers who built a giant 4 x 6-foot color driver's license, stuck their heads through a conveniently placed cutout, and shot the thing with a Polaroid camera. Presto, instant false ID! Today, it's easier to borrow Mom or Dad's scanner. Want to park in the faculty and staff lot at your local university? Your scanned permit may look better than the real thing.

These issues don't need to be discussed here, since they are relatively clear-cut. Scanning and printing your own money is illegal. Altering an image to deceive or generating false documentation can get you in a lot of trouble, too.

However, when you start scanning things for clip art, the rules blur in a hurry. You're not trying to deceive anyone and usually you aren't attempting to deprive another hard-working professional of his or her due income. You just see an image you like and decide to drop it into your desktop publication. After all, that's what scanners are for, isn't it?

However, here are a few guidelines you can use to see if you're on the right track from a common sense standpoint. By definition, anything you capture with a scanner already exists on paper. Text that is read with OCR software was typed or printed out by someone. Photographs were taken by someone with a camera. Line art was created by an artist, even if that artist's skills are limited.

If the creator in any of these cases is yourself, the opportunities for problems are reduced. If the image belongs to someone else, however, you could have problems. You probably won't be sued unless you appropriate someone else's image quite obviously for profit. Scan in a picture of a hobo to illustrate an article on poor investments and probably nothing will happen. Capture a few hundred pages from a clip art book and try to market them as your own clip art library, and you will certainly have problems.

Creating an image yourself is not a guarantee of protection. You can't draw your own rendition of Mickey Mouse and expect to use it, even if you are clever and make a few changes, like giving him 10-fingered gloves. Characters, logos, and other trademarked material are protected even if the actual drawing you scan is not.

You can ask for permission, but realize that Walt Disney makes a lot of money just allowing people to put Mickey's image on things like watches and clothing. The company is, therefore, very unlikely to let you use him for free without a very, very good reason. Nor will a big corporation be liberal with its logos and other elements of trade dress. Once these things lose their protection—as they can through repeated unauthorized use—anyone can use them for anything. No firm wants to see that happen.

The most recent copyright laws protect works for the author's lifetime, plus 50 years. Copyright on works published before 1978 was extended to a total of 75 years. In 1991, you can be fairly certain that artwork published before 1916 has moved into the public domain.

Images can cause legal problems for you in other ways. Even if you take a photograph yourself, you still may not be allowed to use it if the people pictured are recognizable. A special document called a photo release form should be signed by each person in the picture, giving you permission to use his or her image in your publication.Most photo stores have pads of these releases you can use. Note that a photo release offers some protection but is never absolutely ironclad. If you do a lot of scanning of such images, you might want to consult with an attorney who knows the field and who can also draw up a more binding release for you.

When you may not need a release for scanned images:

When the image is for your private use and not for publication. A few snapshots you scan and show to your friends probably won't be considered published.

When a photo was taken in a public place and will not be used in anything that could be construed as advertising. News photographers take pictures of people in the act of being themselves in public all the time and don't bother with photo releases. If your publication qualifies as news, you probably don't need one either.

When people are not recognizable. If faces are obscured, out of focus, or so small as to be unidentifiable, you don't need a release. How could someone prove that the photograph was of him or her? Keep in mind that a person who always wears distinctive clothing or has other distinctive features (he or she may be eight feet tall and have large ears) may be identifiable even though you can't see his or her face.

When the person is a public figure. Generally, public figures such as legislators and entertainers are semi-fair game, unless you've broken some law to gain access to them.

When you do need a photo release:

When the picture will be used in a way that might be taken as promotion or advertising. You can include a picture of Teddy Kennedy in your newsletter as editorial matter, but you can't appropriate his image for an ad or testimonial. If he signs a release authorizing this use, you're off the hook, as long as the picture is used in the ways indicated by the release. Make especially sure you have a valid photo release before using a scanned picture in advertising.When the subject is portrayed in an unflattering or misleading way. You can libel someone with an image. If your picture seems to show Joe Jones stealing a woman's purse – and he wasn't – it may not help you that the photo was taken in a public place. You are particularly open to such charges of graphic libel if your desktop publication doesn't happen to come from a recognized news organization. A photo release can help, if Joe Jones has an open mind.

In the real world, scanner users use images that are not their own in publications. If you are one of them, ideally, you should use a painting or drawing software package to modify these samples sufficiently to make them your own. How much modification is required is open to interpretation.

If your publication has limited circulation, say, within your own office, the chances that anyone will bother to object to your use of his or her image are slim. If it makes the New York Times best seller list, you could have problems.

If you're producing a publication with wide distribution (say, more than 100 copies or so) and you have any doubts about material you're scanning, you may want to check with an attorney who is familiar with copyright issues. The odds are low that you'll be sued for using scanned material, but who wants to take a chance?

Managing Clip Art

We've already talked about the advantages of maintaining clip art in hardcopy form for easy management. File folders or a looseleaf note-book are probably the most convenient housings, since both allow you to shuffle pages around and change the order in which you store them.

Once you've scanned the images and want to keep them in electronic form, you have a more serious problem. There's no way to identify an image using words alone when you have dozens of similar images.

Some image editing software allows you to preview images in low resolution before loading them, but you still have to preview each and every possibility to find the one you want. The same applies to special image viewer utilities. Such a utility may offer an image a bit faster than loading the full file into your application program, but it doesn't really automate the search when you must preview a large number of images.

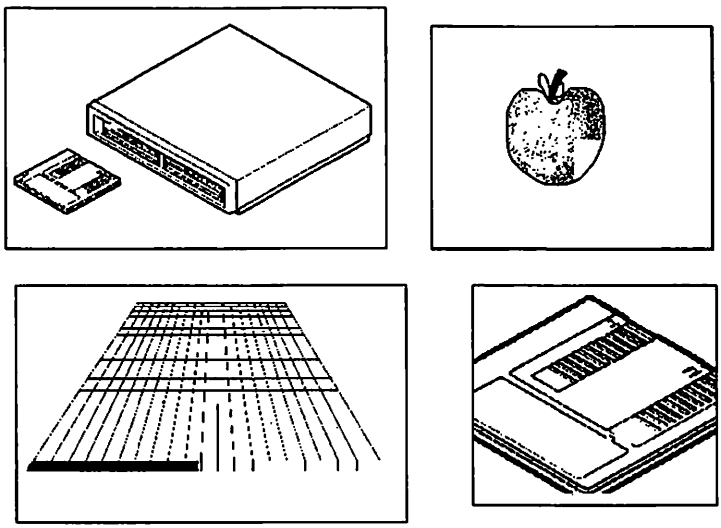
The only real solution is to build your own image catalog. One way I have done this is to load each image and use the shrink tool provided by the image editing software to make a thumbnail image a half inch or so high. I collect 30 to 40 of these and paste them all in a single document that represents all or part of a category. I can print this document or load only the visual catalog to find what I want. The catalog is time-consuming to build, but it does let me search for an image visually. Figure 7.7 shows a sample collection of thumbnails.

How To Scan Clip Art

As you might expect, clip art is scanned like any other image you may work with. There are some special considerations that apply to this type of artwork, however.

- If you can't or don't want to break apart a book to scan in a piece of clip art, consider photocopying the page and scanning the duplicate; high quality copiers don't lose a lot of detail. You can also try using a flatbed scanner with a large bed to allow positioning the book.
- Scan line art at the highest resolution available. It may look sharp on the printed page, but you'll be able to use the extra resolution later on to restore detail that was lost during the reproduction process. An exception to this rule would be if the line art is of high quality and you

Figure 7.7



Sample thumbnail illustrations

already know what size you'll need for your publication. Then go ahead and scan at the final resolution.

- If copying yellowed old pages, experiment with using yellow acetate filters of various densities to improve contrast. Yellow makes blue ink (as on old ledgers) appear darker, too. Other colored filters can lighten tones of their own color, while darkening their complements.
- Capture a group of images with one scan to save time. If you need more than one image on a page, you can scan the entire page at a high resolution. This helps you grab a larger number of images in a short time, which can be important when you're using someone else's reference material as your source. Later on, you can load each file into your

image editing software and save the individual images in files of your own. That reduces the disk storage requirements for the collection.

- Consider storing small clip art files in one larger document. Logos and little "bugs" used to add interest to a publication take up a minimum amount of disk space no matter how small they are. Putting 20 of them in a single 32K file, say, can save you 40 or 50K of disk space.
- If the paper you are scanning is translucent and has printing on both sides, putting a dark piece of paper behind the page being scanned can absorb extra light that may bounce back and "print through."

Summary

Desktop publishers may find that digitizing clip art is one of the most useful applications for a scanner. Users who lack art skills can get good quality generic art without paying an artist to draw it specifically for a publication.

Traditionally, clip art has been black-and-white art published in hardcopy form for use by other graphics professionals. Art is printed in several sizes on one side of glossy paper, so the images can be clipped and pasted directly on a finished mechanical.

The photographic equivalent of clip art is the stock photo. These generic photographs must be halftoned before use, and the fees to use them are higher, varying by the application.

Clip art can provide the visual you need or form the basis for completing a finished drawing on your own. While clip art is available in digitized form, scanner owners can save money by capturing their own.

If you keep hard copies of your clip art and scan it as you need it, you won't be scanning images you don't eventually use. You also gain flexibility in resolution, sizing, and scaling to fit the specific use. You save disk space by not storing images that aren't being used, and browsing through hard copies is considerably more convenient.

On the other hand, prescanned art gives you quicker access, the ability to re-use an image without rescanning it, and additional filing flexibility. If you modify an image, you'll want to save the electronic version to avoid duplicating the work the next time you use that image.

There are several sources of clip art. These include commercial clip art books, material on which the copyright has expired, public domain work such as that published by the government, and copyrighted works which you can get permission to use. Don't forget the storehouse of images in the reference departments of businesses, public libraries, universities, and schools.

If you can't manage clip art in hardcopy form, you can use various utilities to help you provide your electronic images with more descriptive file names. Probably the best, although somewhat time-consuming, method is to prepare a miniature thumbnail view of each image, which you can browse through visually to select the image you want.

8

Manipulating Scanned Images

If all you wanted to do was duplicate an image in a desktop publication, you really wouldn't need a scanner. Any artwork you can find, even previously screened halftones, can be duplicated photographically, enlarged or reduced to the correct size, and then inserted in your layout for offset printing.

Your scanner allows you to modify the images you capture. In some cases, you may just want to clean up a piece of art that has collected some artifacts or broken lines during the scanning process. I quite frequently scan logos for use in newsletters. I usually do it at a fairly high resolution so I can later use an image editing program to make sure all the straight lines are straight and the curves are smooth.

In other cases, the scanned artwork may be a starting point for a more complex piece of art. You can combine images from different sources, add your own modifications, and produce an entirely new work. If you're working with a gray scale or color image, you can move, delete, or add elements, and reverse portions of the image.

These are powerful capabilities, which are changing the way we look at images. As recently as 150 years ago, seeing was believing. Special skills were required to falsify what we thought we saw with our own two eyes. Illusions were the province of professional magicians, artists, and criminals, all of whom knew we were willing to pay for the privilege of being fooled.

Our well-founded trust in visual information was shattered by the introduction of photography. The first movie-goers sometimes fled the theater when an onrushing train was shown on the screen. French magician George Melies astounded audiences by using a camera fixed to a tripod to film a woman. At one point, he stopped the camera, asked the woman to leave, and then restarted the camera. To viewers accustomed to believing their own eyes, the disappearance was nothing short of a miracle.

Opening Pandora's Box

Until recently, you could believe what you saw in photographs, too. Even the best camera tricks and darkroom magic could be unmasked by an expert. Airborne hubcaps and Frisbees might have looked like flying saucers at first glance, but close scrutiny usually showed them for what they were.

Things are no longer what they seem. A digital image can be manipulated in ways that are impossible to detect. Modifying line art at high resolutions is a trivial exercise; anyone with a few minutes' training can manipulate individual pixels. A bit more skill is required to modify continuous tone or color images, but anyone with a color-capable Mac, a scanner, and the proper software can learn to do it.

So, we no longer can believe our eyes. If anyone can change an image, anyone may. A professor of electronic imaging at one of our leading photography schools told me that he hoped special image editing software would be developed with only a limited set of features. Such programs would be used at news organizations to selectively adjust contrast, correct color, and crop photographs. More advanced capabilities would be deliberately left out to eliminate the temptation of image tampering.

The gelding of such software would probably be nothing more than an ineffective attempt to close Pandora's box. Couldn't adjusting the color so an event appears to take place at dusk instead of high noon be considered tampering? It might if the event were an innocent stroll on the beach by a political figure and an associate of the opposite sex. And who's to say the news gatherers might not have their own software hidden away somewhere? We've already seen the Great Pyramid moved a few hundred yards to produce a better cover for National Geographic. What's next?

Most likely, the next step is for the average computer user to buy a scanner and learn exactly what can and can't be done to manipulate images. That's the goal of this chapter.

Types of Images

Scanners and associated software can produce several different types of images. Each type can be manipulated in different ways, and each carries its own advantages and disadvantages. Let's look at the various classifications first.

Bit-Mapped Images vs. Vector Images

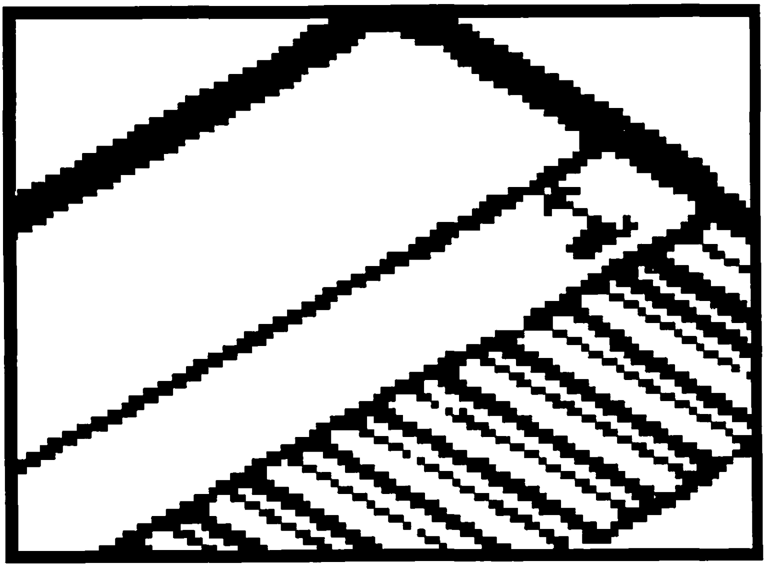
Scanners always capture an image dot by dot and line by line, producing a bit map. The computer doesn't need to know anything about the original image. It makes no difference whether the artwork is a color photograph of your house or a blueprint of the floorplan. The scanner looks at each pixel in turn and reports how much light was reflected or transmitted to the sensor.

Therefore, you have considerable flexibility in choosing the amount and kind of information you want to capture. If you are interested only in black or white information, you can specify that with the scanner controller software. If you'd rather have 16, 64, or 256 levels of gray or full color and your scanner is capable of that, you can do that, too.

However, bit maps can't be resized after scanning without undesirable results. If you scale an image up, all you are doing is enlarging the original pixels used to capture the image. Eventually, the pixels become visible, producing a mosaic effect in continuous tone art and infamous "jaggies" in line art. Figure 8.1 shows what happens when you enlarge bit-mapped artwork.

Scaling an image down to a smaller size requires throwing away useful information. If you want to enlarge the image later, you must return to the original file; the data is no longer contained in the shrunken

Figure 8.1



As you enlarge a bit map, the individual pixels that make up the image become more obvious. Diagonal lines produce an effect called "jaggies."

bit map. Enlarging or reducing a bit map to a size that isn't an even multiple of the original resolution is also problematic. Double the size of an image, and each pixel becomes four pixels in the new picture. Reduce it by half, and one pixel must represent the spatial value of four others. But change an image by some odd increment, and these neat relationships vanish. You might need to enlarge a 3-inch picture to 4.5 inches. Can the computer use 1.5 pixels in the new image to represent every pixel in the old one? Not easily. The compromises that must be made inevitably lead to reduced quality.

A second disadvantage of bit-mapped images is that the more detail you want to capture and retain, the higher the storage requirements. A 75 dpi binary image requires 703 bytes of storage per square inch; at 300 dpi you need more than 11K of disk space per square inch. Scanners with 600 dpi capabilities will be common before the end of the decade, and a single square inch of black-and-white information stored at that resolution will require 45K. A 256-gray level image at that resolution calls for 360K per square inch, and a color scan requires more than a megabyte for a single inch-square scan!

The most disturbing thing is that these requirements are the same whether the image is full of detail or consists mostly of white space. Since the computer knows nothing about the image, it must capture all of it to be sure the important information is retained.

The previous discussions of pixel depth have acquainted you with the various types of bit-mapped artwork. I'll list them here as a summary:

- **Line art.** This is a binary, black-and-white image captured in 1-bit mode. Such artwork consists only of lines, dots, and other monochrome images. Text, drawings, and similar images are all treated as line art.
- **Gray scale images.** If more than one value of darkness is retained by a system, the image is said to have a gray scale. The scale may be very limited. For example, a 2-bit system would be able to store only four levels of gray (actually, three plus white). Figure 8.2 shows the number of gray values possible at various pixel depths.

  In the real world, the most common scanners are 1-bit, 4-bit, 6-bit, 8-bit, and 24-bit models. Some hand scanners advertise the ability to capture 32 tones. All 24-bit scanners are color scanners. At the high end, there are some scanners that capture extra bits of information (this is called oversampling), which is then reduced to the amount of data required.

  For example, since an 8-bit scanner may in truth be sensitive enough to detect only 7 bits of information, some scanners capture 12 bits of data and then interpolate it down to the 8 bits that are desired. With 4096 different values to work with, it's simple to calculate 256 tones from the extra information. The same process is used with high end color scanners, although in some 32-bit systems, the extra 8 bits are actually used to represent the degree of transparency an image should have when overlaid on another.

Figure 8.2

Bits	Tones
1	2
2	4
3	8
4	6
5	32
6	64
7	124
8	256
12	4096
16	65565
24	16.7 million
32	4 billion

This chart shows the number of gray or color tones that can be represented with a given number of bits.

If you have enough tone values, an image will appear to have an infinitely small progression from black to white, so it will be said to be a continuous tone image. However, even black-and-white photographs don't really have continuous tones, because they are made up of microscopically small silver crystals that have been developed to a black color against the white background of the substrate.

- **Color images.** These are continuous tone images in which three gray scales are retained to represent the amount of light reflected in each of the three primary colors. If you were to look at an individual color plane of a 24-bit image, it would look a lot like an 8-bit gray scale image. By tinting each of these three gray scales and superimposing them, we can reproduce the original colors.

All three types of bit-mapped graphics are manipulated using paint type software. Such programs allow you to work with bit-mapped images at various levels of resolution and pixel depths. Some handle only 1-bit or 4-bit files, while others can accept full 256-tone gray scale or color images.

Vector Art

The other kind of image is the vector graphic, which is outline-oriented. That is, the image is stored in the form of descriptions of the lines and curves that make up the image, rather than as a bit map representing the image itself.

Vector graphics are actually closer to the way humans construct drawings than bit maps are. A square is drawn by joining four lines of equal length at 90° angles, for example. Computer assisted design (CAD) software and drawing programs like Adobe Illustrator and Aldus Freehand use this approach.

While scanners don't produce vector-oriented files, your bit maps can be converted to outlines using a program with autotracing features. This capability is discussed in more detail in the next chapter.

Vector graphics have their own advantages and disadvantages. Vectors can't be used to represent continuous tone images, and for that reason your gray scale and color images won't convert easily to vector format. Draw programs are best used to manipulate traced scans of line art and to originate new line drawings.

Outline art can be stored on disk more efficiently, because only the descriptions are retained. If a page is very simple, the file is tiny compared to a bit-mapped image of the same page. However, this advantage begins to lose its luster as the page grows in complexity. While a bit map file remains the same size at a given resolution no matter how much detail is present on the page, vector files grow larger. At a certain point, a vector file can actually be larger than the equivalent bit map.

The important advantage of vector art is its scalability. You can double or quadruple the size of an image with no loss of resolution. Your system just recalculates the correct length and width of each line based on the new size and draws the image correctly at full resolution.

Image Editing User Interfaces

Later in this chapter I'll cover the key tools available in both bit-mapped and vector image editing tools. But first, I'll explain some of the differences you're likely to encounter at a somewhat higher level: the user interface. To be blunt, some programs suffer from a clumsy, difficult-to-learn and/or difficult-to-use interface that hinders their utility That's in spite of the fact that the Macintosh supposedly has an interface that is more or less standardized among all its programs.

Does your program's interface allow loading multiple image files into sizable windows? ColorStudio allows loading many different image files into separate windows at one time. Then, it goes one better: you can create different *views* of a single image which you can look and manipulate. For example, you might want to have a 100 percent view of a particular image, in order to view the entire subject. A second window at 800 percent would allow you to manipulate individual pixels.

How can you change the scale of the image within a window? Some programs let you click on the mouse button to make the magnification progressively greater or smaller. Other interfaces make you find a magnifying glass icon and fiddle with it.

What does double-clicking on a tool do under a particular interface? With some, it provides an enhanced version of a command; double-clicking on the eraser may erase the whole screen. With other interfaces, a double click brings up a menu of additional choices, such as brush size or palette color. Some programs have a mixture of double-click functions.

You'll see additional comparisons of interface functions in the next sections of this chapter, in which I discuss some of the most popular tools found in both bit map and vector-based image editing software.

Bit-Mapped Image Editing Tools

As you work with scanners, you'll need to use both paint and draw image editors to achieve the effects you want. Each type of software has a range of tools that allow you to modify, clone, and convert scanned images. The next section introduces you to some of those tools. I'll explain only the basic tools found in most paint and draw programs in this chapter. More advanced tools used for color correction, adjusting brightness and contrast, and creating halftones are discussed later in the book.

Selection Tools

The ability to select a portion of a scanned image for manipulation is a key capability, and most image editing software provides at least two methods of doing this. A selection rectangle allows you to define a square or rectangular area of the image for manipulation. You position the mouse at one corner of the area and click, then drag the outline until you've defined the other corner. Thereafter, you can carry out various functions on the portion of the image within the rectangle (or, with some software, on everything outside the defined rectangle).

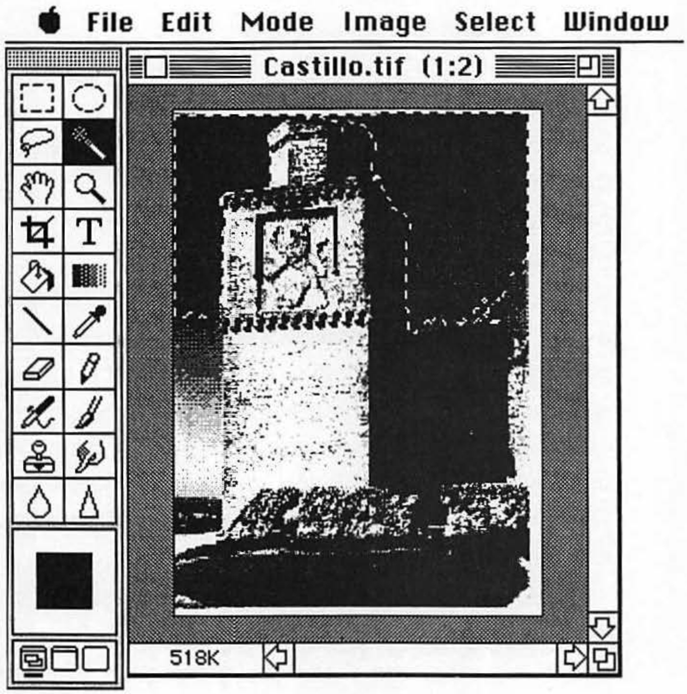
The second way of defining an area is with a scissors tool. The scissors, as you might guess, are used to define an irregularly shaped area. Some packages use other analogies, but the concept is the same. You can draw a freehand line to define the working area. Some packages require you to click once to define each point of the area. This provides greater accuracy but slows you down if all you want to do is draw a roughed-in, irregularly shaped area. Curves, for example, must be drawn by clicking the mouse many times.

Many programs provide even more advanced selection tools. You can specify a particular color or range of colors for pixels that define the area, so you can easily select a portion of an image that contains similar tones. The magic wand found in Adobe PhotoShop is an example of this sometimes amazing tool. I've used the wand to neatly remove a figure from a scanned photograph, leaving a hole in the background that could be easily filled in with neutral scenery. As color becomes more prevalent, you'll find more programs that let you work with and select each color layer separately.

The things you can do with a selected area vary by program. All allow you to move the selected area to another portion of your image, duplicate the area, and cut an area out of an image entirely. You may also be able to copy the area to a disk file, flip it horizontally or vertically, rotate it, slant it, or elongate it in one direction or another to produce a squashed or stretched image.

Other common functions include inversing (reversing the tones to produce a negative) and shrinking or growing the selected area to make it larger or smaller. A typical selection tool in use is shown in Figure 8.3.

Figure 8.3



A selection tool at work, the Magic Wand from Adobe PhotoShop.

Selection Tips

- If the area you want to select is defined by a solid tone, try painting the tone with white before starting to select the part of the image you want. For example, a head and shoulders portrait may have been taken against a plain black background. Pouring white paint into the background area can eliminate it entirely, leaving the head and shoulders of your subject cleanly defined for selection.

- If your image is a gray scale scan with 64 or more tones, a background that appears black may actually be made of four or five very dark grays. You can still remove them all. Pour your white paint in turn into each of the blotches of tone that remain. If a few pixels of a tone appear along the boundaries of your subject, fill the entire background with that color, then replace it with white. If the same tone appears at the edge of your subject, it may "eat" into the image itself. You can clean this up at the bit level later on.
- Selecting one portion of an image is a quick way of cleaning up defects in another portion. Dust spots can be covered up by selecting an area of similar tone and pasting it over the spotted area.
- You can also use this technique to add windows to blank walls, remove or add trees, and otherwise modify an image.

Painting Tools

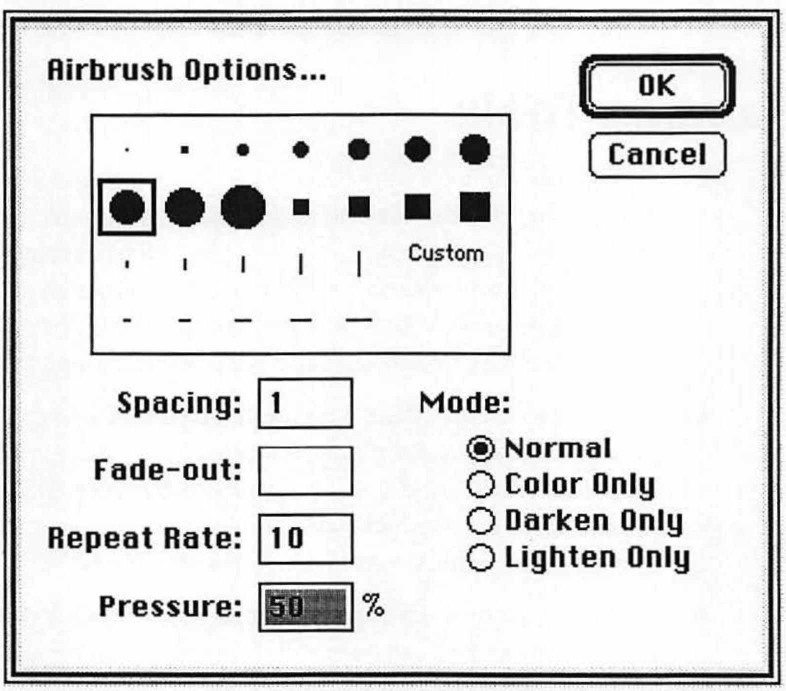
The painting tools allow you to add or change the tones of an image. For line art, you can use one of the line drawing tools described below, brushes, pencils, airbrushes, or other tools. All paint using black or white. Gray scale or color images require more care, because you must select the tone to be painted with as well as the tool. Clumsy painting can quickly ruin an image, so be sure to save a copy before starting to work.

- Pencils and brushes paint with the selected tone using widths from one pixel to as many as 64 pixels at a time. Some software also gives you patterned brushes and a selection of brush angles. Brushes are useful for editing at the pixel level and for laying down large swathes of tone. A white brush makes a good, custom-sized eraser.
- Spray cans, airbrushes, and similar tools lay down a random spattered pattern of tone. This can be useful for blending boundaries and for filling areas with irregular tones.

- Some packages provide an eyedropper tool that can be used to pick up the exact tone you want to paint with. This is an essential capability when working with gray scale and color images that may have 256 or more individual tones that are difficult to differentiate by eye.
- Erasers are nothing more than white paint brushes. Some packages offer color erasers, which allow you to remove only selected shades or to replace one tone with another.
- Shape drawing tools are a type of painting tool that draws fixed types of filled or hollow squares, circles, irregular polygons, ovals, rectangles, lines, curves, and so forth. They can be drawn or filled with the current selected tone or pattern.

Some packages provide filters that can be used to modify portions of an image using either brush tools or the selection tools. You can smudge or blend a line, add a tint, blur or sharpen part of an image, change the

Figure 8.4



The airbrush tool shown here is one of the many variations on the basic paint tool that you'll find in image editing software.

brightness or contrast, reverse, or perform other transformations on small portions. Figure 8.4 shows some painting tools.

Zooming In and Out

The zooming—viewing an image in enlargement or reduction—capabilities of an image editing package can determine how flexibly you can edit an image. You'll need to enlarge a section as much as possible—say 8 to 10 times—to work with individual pixels. Other times, you'll want to look at larger sections in more detail. The best packages provide multiple levels of zooming—2X, 3X, 4X, 6X, and 8X, for example. These are also listed as percentages: 200%, 300%, 400%, 600%, 800%, and so forth.

Zooming out lets you view an image that may be too large to fit on your screen at 100% magnification. You should get 2X (50%), 4X (25%), and 8X (12.5%) zoom out as a bare minimum.

The very best packages allow you to perform all editing functions at any zoom ratio and to have several views of a single image on the screen at one time. Some accomplish this with a split screen, which lets you split the screen horizontally or vertically and view a portion of the full image in one half while your zoomed section is shown in the other. Other packages show a single image in several windows, like ColorStudio. You can edit an active window while viewing another window for reference.

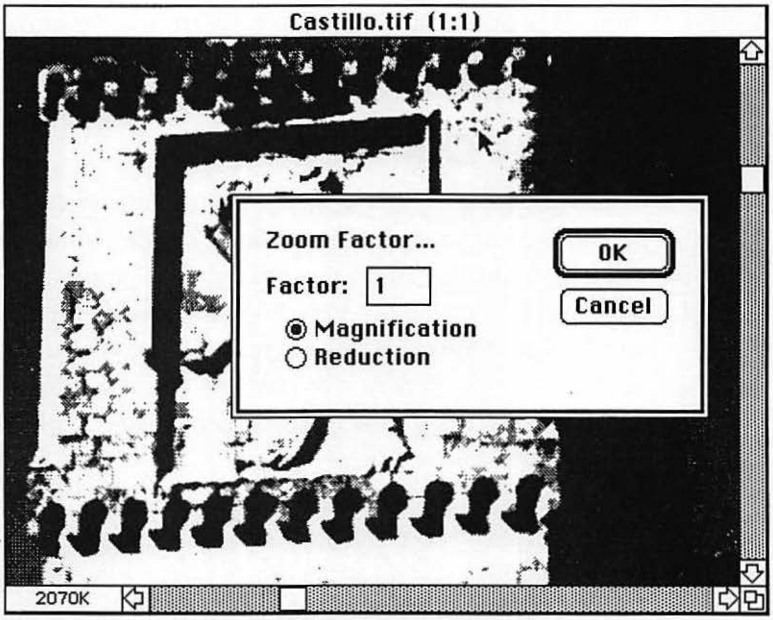
If I can't get the exact zoom ratio I want, I sometimes enlarge the entire picture using the sizing and scaling capabilities of the package. Some programs have a scaling option. The image can be enlarged 110, 120, 140, or 150%, modified, and then rescaled to a final size.

Zooming is shown in Figure 8.5 on the following page.

Other Effects and Tools

Since you may be limited to 256 or fewer tones on the screen at one time, the ability to edit or exchange palettes can be important. Most image editing software provides a selection of palettes that can be

Figure 8.5



Zooming functions allow you to work with an image up close.

recalled from disk. They may be evenly distributed gray scales, color palettes with extra grays, flesh-oriented palettes, or palettes with emphasis on bright neon colors.

The text tools allow you to add text to scanned images, using any of the supplied bit-mapped fonts. In most cases, your choices are limited, and the resolution of the characters may not be acceptable. You're better off using your desktop publishing package to add text once you've finished editing an image and have placed it in your final document. You'll soon see image editing programs with special text layers that allow text to be edited separately from the rest of the bit map.

Nearly all image editing packages let you paste in other images that you store on disk. The best also provide options on the transparency of the pasted-in image—should it blend with the existing image, cover it completely, or only blend around the edges? You should also be able to rotate, distort, and scale an image you are pasting.

We're also seeing software that allows you to create three-dimensional shapes and then apply images to the surfaces of those shapes. You can therefore make an object into a sphere, cube, or other shape, starting with a flat, two-dimensional scan.

Improving on the Original

Bit map image editing software can actually allow you to improve on an original image by scanning less-than-perfect artwork and then fixing it up later using your computer tools. The following are some of the functions that are particularly useful for enhancing gray scale and color images:

- **Sharpening.** Many packages have a command for sharpening all or portions of an image. This is done by increasing the contrast along the edges of objects in an image. The software determines the edges by looking for relatively sharp transitions between tonal ranges. In an unsharp image, the transitions are blurred. By increasing the contrast, apparent sharpness is increased, although no new image information has been added.

  When you use this feature, you'll usually be offered the option of using mild, intermediate, or high sharpening. I've found the highest degrees of sharpening produce grainy effects that are probably not what you are looking for. Start out with mild sharpening and upgrade if you think you need more. If your software makes you specify how large an Undo buffer you want, make sure you allocate enough memory to revert back if you don't like the results.

- **Contrast/brightness adjustment.** I've found that it's often better to scan a slightly flat, low contrast image that is full of detail than one that appears normal to the eye. You can frequently adjust the contrast of the final image to compensate, giving you the opportunity to decide after the scan which details in your image are important.

  However, don't be put off if the only available copy of an image appears dark and muddy. I've been amazed at the detail I've been able to pull out of overexposed pictures just by adjusting the contrast and darkness after the scan.

- **Color correction.** We'll be looking at color in more detail later. However, you should keep in mind that an image that appears to have a color cast to it can be corrected during color scanning.

Drawing Tools

The distinctions between bit-mapped and vector-oriented image editing software are blurring in the Macintosh world. Programs like Canvas offer both types of layers that you can have in a single document, simultaneously. You can also bring bit maps into drawing programs and then trace them to change the outlines of the scan into a vector image. Some outline drawings can be exported to bit-mapped editing software.

The trick is to assign a group of layers to each portion of an image. Some can be bit-mapped layers, while others are vector layers. You can choose to view and print any or all of them.

Here, we'll concentrate on what you can do with the images once they've been converted to outline format. The key advantage is that you can enlarge or reduce the image as much as you want, without losing resolution.

Object Orientation

Vector programs look a little like their bit-mapped counterparts on the surface, using various tool analogies to represent individual functions. In practice, however, they are used quite differently.

For example, line and shape drawing tools produce independent objects that can be moved, enlarged, deleted, or manipulated even if you later add other lines or shapes on top of them. To be joined, individual objects must be grouped, using the grouping capabilities of your software. Each object occupies a specific layer, so you can send some objects behind and bring others to the foreground if you wish.

Vector objects aren't painted as bit-mapped shapes are. Instead, you fill them with patterns or tones that simulate a bit-mapped tone effect.

Actually, the software provides instructions for filling the shape so the pattern won't become apparent if the shape is enlarged or printed at a greater magnification. The software may actually use PostScript instructions to produce fills, gradients, and other effects.

Fill patterns are usually measured in percentages of a given tone, because dots of varying sizes are used to print them (you'll want to read up on halftoning later in this book). That is, a 40% fill contains 60% white space. Some 40% of the area will be filled with the gray tone or color you are using for the fill.

Gradients, or fountain fills, change color from one tone to another gradually, producing a shaded effect. You can specify a start color and percentage, and then a final color and percentage. That is, you may start with a relatively light 10% cyan and finish the gradient with a dark 90% blue.

You can specify an angle for the gradient and whether the fountain operates in a linear or radial direction. A radial fountain used on a circle produces a shaded, spherical effect, for example.

Greater Flexibility

Because they allow you to work with outlines rather than bit maps that must be scaled up and down in fixed increments, vector drawing programs are usually much more flexible in some areas. For example, you can usually zoom over a much greater range of magnifications, and fractional ratios are absolutely no problem. The usual tool is a sort of magnifying glass that you use to select a rectangle representing the area you want magnified. The software then enlarges that portion to fill the screen or active window.

For the same reasons, the text capabilities of vector drawing programs are more advanced than those found in bit map editors. You can draw from the same kind of outline technology that produces scalable PostScript fonts, even if you don't have a PostScript printer.

Your Drawing Toolkit

As I've noted before, this book is not intended to be a comprehensive review of all the hardware and software of interest to scanner users. For a broader view of products, I point you toward *MacUser, MacWorld, InfoWorld*, or my companion volume, *The Complete Scanner Handbook, Macintosh Edition*. However, I have taken some time to look at several of the latest offerings in each category and can recommend several industry leaders that are worth your consideration. This next section discusses a few of the leaders in each category.

Bit-Mapped Software

The differences between color and gray scale painting and image editing programs are blurring. So-called low end painting programs, like UltraPaint, can manipulate full-color images nearly as well as some high end products. You'll want to look carefully at the capabilities of each software package to select the one best for your needs.

All gray scale and color software will let you perform painting tasks, using pencil, paintbrush, and eyedropper tools. Image retouching programs add the ability to work with different colors as separate layers or channels. This capability is essential for color correction and color separation.

Adobe PhotoShop

Adobe PhotoShop built up an early lead in the race for dominance of the color image editing arena, based on an excellent mix of features, price, and availability. PhotoShop can do most of what needs to be done with continuous tone color images. It was originally priced at less than half the asking fee for LetraSet's ColorStudio, and it was furnished free with a number of important color scanner products.

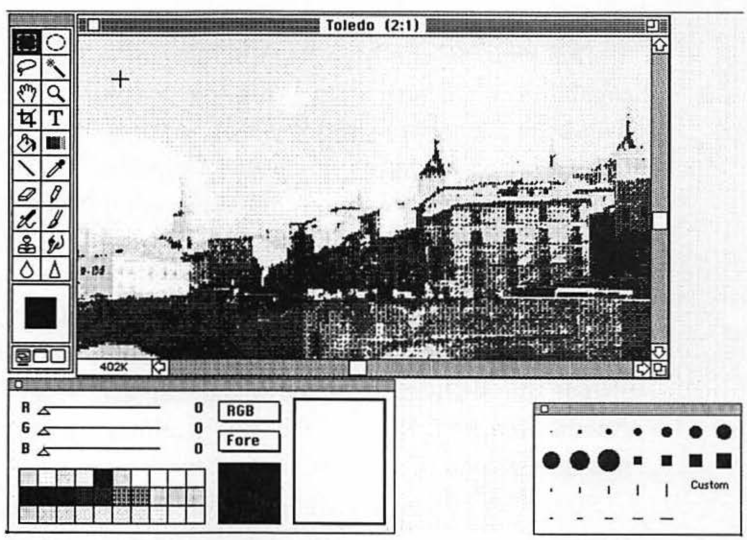
Now ColorStudio is comparably priced, and is a key rival in the OEM software arena. It was furnished with my Epson color scanner, for example. Yet, PhotoShop continues to set standards for features and performance.

For image editing, PhotoShop's strengths include an extensive set of image filters. These can be used to sharpen edges, to blur portions of an image, or to perform other modifications.

PhotoShop lets you manipulate each color layer of an image separately. It supports 32-bit color, so the alpha channel can be used to hold text, or store transparency information about a portion of an image. You can control brightness, contrast, and color balance—something that paint-type programs don't always allow, even if they can handle 32-bit color.

PhotoShop supports GIF, TIFF, PICT, PIXAR, and Scitex CT formats. It has all the expected painting features, including an editable paintbrush, airbrush, and paint bucket. The program is shown in Figure 8.6 below.

Figure 8.6



Adobe Photoshop, one of the most advanced image editing programs available today.

The only glaring omission is the lack of a drawing layer for object-oriented graphics. You'll have to export your PhotoShop files to another program to do any work with PostScript text or images.

Studio/32

I used Studio/8 in preparing my last scanner book, and so I found this 32-bit upgrade exceptionally easy to learn. The earlier program remains in the line for those of you who have only 256-color capabilities.

However, those with 8-bit color boards still may be interested in Studio/32. It allows you to open and edit 16- 24- or 32-bit color images no matter what type of color display system you have. Studio/32 will automatically choose the right 256 colors to show the image on an 8-bit monitor.

As with more color editing programs these days, it uses virtual memory techniques to allow manipulating images that are larger than your available RAM space. Your hard disk will be used to emulate more memory as needed.

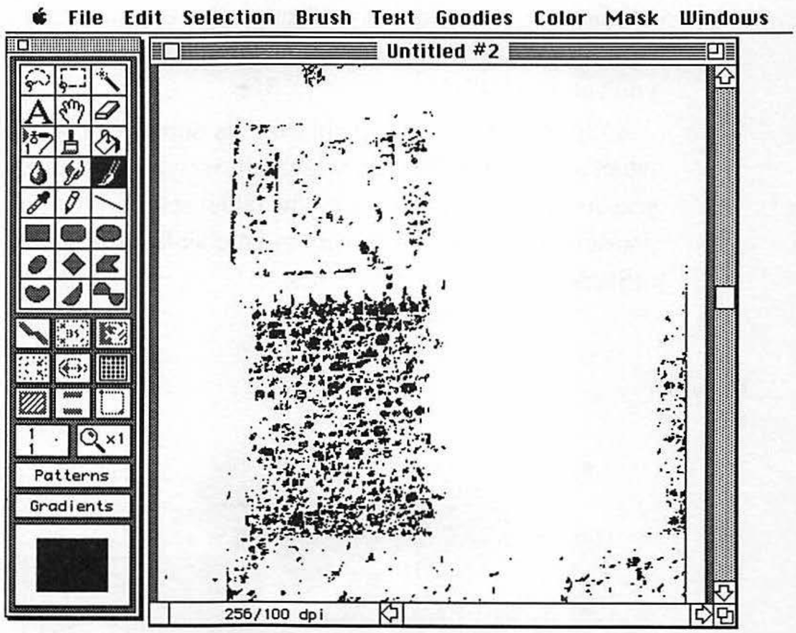
This painting and drawing program doesn't have color separation capabilities. You have to save your files as color TIFF files and separate them with a tool like Aldus PrePrint. Alternatively, you can load them into an image editing program like ColorStudio. You might actually want to do this if you value Studio/32's very robust image editing tools, and have some other use for ColorStudio, such as driving your color scanner.

Selection tools include or exclude specified colors. There is a Bezier tool for adding curves. Studio/32 has an intriguing feature that lets you change the perspective of the image using x, y, and z axis controls. Another innovative tool is the spherization capability, which wraps a two-dimensional shape around a sphere.

Studio/32 has an outstanding PostScript text layer. You can also manipulate text as bit-maps. Those who must sort through dozens of images will appreciate the package's Preview feature, which lets you scan thumbnails before deciding which file to load.

Studio/32 supports MacPaint, PICT, and TIFF file formats. Studio/32 is shown in Figure 8.7.

Figure 8.7



Studio/32, one of the most sophisticated painting programs.

ColorStudio

It's dangerous to tout any one feature, since by the time this book is published, rival programs like PhotoShop may have it too. But, at this writing, if you want a full-featured image scanning/editing program with a drawing layer, ColorStudio is the one.

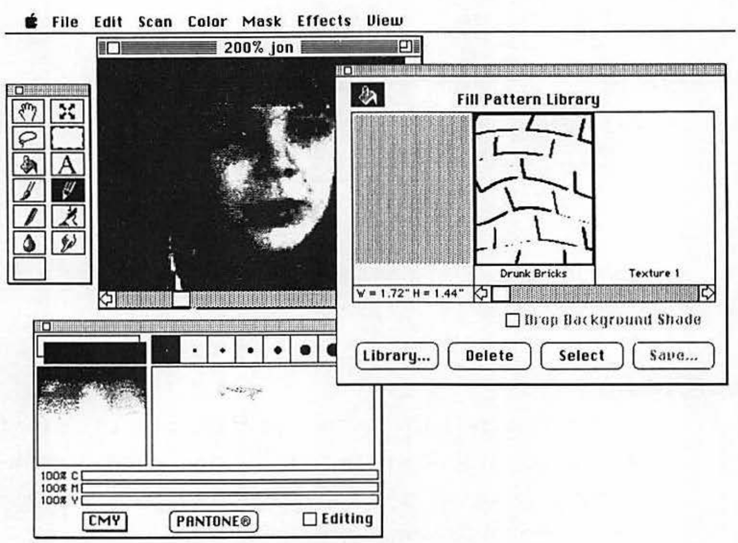
As with other software, you can work with a color layer for retouching and image enhancement. There is also a mask layer, a monochrome layer that can be used to block off or modify portions of the color layer.

An add-on called Shapes provides ColorStudio with a drawing layer. Shapes allows importing Encapsulated PostScript images, but not PostScript text. So, you can't edit text as you can with Studio/32.

ColorStudio has an interesting 'Tween function which creates transitions between a beginning figure, say, a square, and an ending one, such as a triangle. A smooth transition between the two is created for you automatically.

ColorStudio is also notable for its support of Wacom's digitizing tablet and wireless stylus, which allows you to draw electronically as you do with a pen and paper. The tablet responds to varying degrees of pressure, providing those with artistic skills with a valuable tool. ColorStudio is shown in Figure 8.8.

Figure 8.8



ColorStudio is bundled with many popular color scanners, such as the Epson ES-300C .

PixelPaint Professional

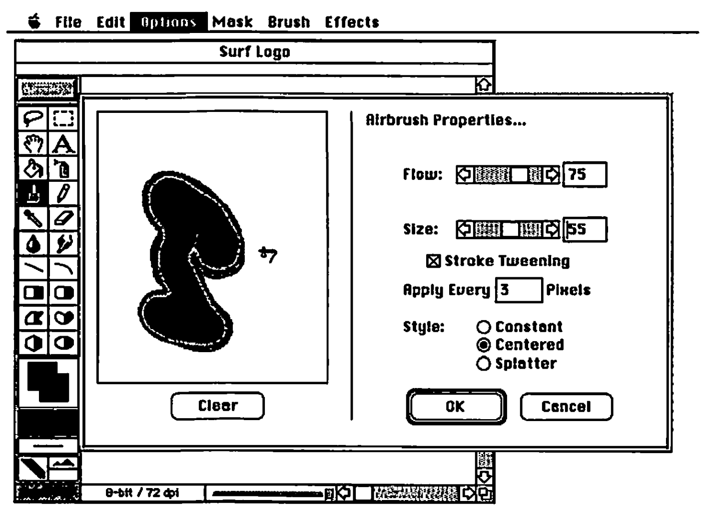
This isn't simply an improved version of PixelPaint. PixelPaint Professional in its Version 2.0 incarnation has a new interface, and a group of important new features.

For example, you can create a masking tool from any other tool or even a portion of an image. There is a new Wet Paint feature that lets you move and edit a shape or text before merging it with the rest of the image. A new gradient editor lets you create custom fills.

Scanner users will be excited about the PixelPaper feature, which makes it possible to add texture to portions of an image. You can now can scan patterns that you want to use as a surface texture.

PixelPaint Professional supports MacPaint, TIFF, PICT, and PixelPaint formats. PixelPaint is shown in Figure 8.9.

Figure 8.9



PixelPaint Professional is an advanced image editing program that has handy features, such as a gradient editor that lets you create custom fills, and a customizable airbrush like the one shown here.

Vector-Based Software

Some day, drawing type programs won't exist as a separate category. The only thing you'll need will be a package capable of handling 32-bit color images in one set of layers, and vector-oriented objects in another set of layers.

We're already seeing programs like that. Canvas 3.0 has powerful bit-mapped and drawing capabilities. Meanwhile, the software packages that started life as draw programs just keep getting better and better.

Adobe Illustrator and Aldus Freehand are available in new versions, while Claris has introduced MacDraw II as the latest edition of the vector-oriented program that started this whole category in the Macintosh world.

Draw-only programs are useful for scanned images only if they have good autotrace capabilities. Otherwise, you can import bit-mapped images and add object graphics to them, but can't change the actual bit map in any way.

Summary

Your scanner allows you to modify the images you capture. At times, you may just want to clean up a piece of art, while in other cases the scanned artwork may be a starting point for a more complex piece. You can combine images from different sources, add your own modifications, and produce an entirely new work. If you're working with a gray scale or color image, you can move, delete, or add elements, and reverse portions of the image.

Until recently, you could believe what you saw in photographs. That's no longer true. A digital image can be manipulated in ways that are impossible to detect.

Scanners produce several types of images, but they always capture an image dot by dot and line by line, producing a bit map. These can't be resized after scanning without undesirable results. A second disadvantage of bit map images is that the more detail you want to capture and retain, the higher the storage requirements. There are three categories of bit-mapped images—line art, gray scale images, and color images.

The other kind of image is the vector graphic, which is outline-oriented. The image is stored in the form of descriptions of the lines and curves that make up the image, rather than a bit map representing the image itself.

Vector graphics are actually closer to the way humans construct drawings than bit maps are. While scanners don't produce vector-oriented files, your bit maps can be converted to outlines using a program with autotracing features.

Vector graphics have their own advantages and disadvantages. Vectors can't be used to represent continuous tone images, and for that reason your gray scale and color images won't convert easily to vector format. Draw programs are best used to manipulate traced scans of line art and to originate new line drawings.

Outline art can be more efficient to store on disk, because only the descriptions are retained. The most important advantage of vector art is its scalability. You can double or quadruple the size of an image with no loss in resolution. Your system just recalculates the correct length and width of each line based on the new size and draws the image correctly at full resolution.

As you work with scanners, you'll need to use both paint and draw image editors to achieve the affects you want. Each type of software has a range of tools that allows you to modify scanned images.

Bit map editors rely on tools like the selection box and scissors, which let you select a portion of a scanned image for manipulation. Once the image is selected, you can move the selected area to another portion of your image, duplicate the area, or cut an area out of an image entirely. You may also be able to copy the area to a disk file, flip it horizontally or vertically, rotate it, slant it, or elongate it in one direction or another to produce a squashed or stretched image.

Other common functions include inversing (reversing the tones to produce a negative) and shrinking or growing the selected area to make it larger or smaller.

Painting tools allow you to add or change the tones of an image. You can use line drawing tools, brushes, pencils, airbrushes, and other tools. Shape drawing tools are a type of painting tool that draws fixed types of filled or hollow squares, circles, irregular polygons, ovals, rectangles, lines, curves, and so forth. They can be drawn or filled with the current selected tone or pattern.

Some packages provide filters that can be used to modify portions of an image, using either brush tools or the selection tools. You can smudge or blend a line, add a tint, blur or sharpen part of an image, change the brightness or contrast, reverse, and perform other transformations on small portions.

Zooming in and out enables you to work with a small portion of an image or to view an entire image that is too large to fit on the screen at 100% magnification.

Bit map image editing software can actually improve an original image, allowing you to scan less-than-perfect artwork and then repair it later using your computer tools.

You can sharpen images, change the contrast and brightness, and correct colors flexibly.

Vector-oriented drawing programs have their own set of tools. For example, line and shape drawing tools produce independent objects that can be moved, enlarged, deleted, and manipulated even if you later add other lines or shapes on top of them. To be joined, individual objects must be grouped using the grouping capabilities of your software. Each object occupies a specific layer, so you can send some objects behind and bring others to the foreground if you wish.

Vector objects aren't painted as bit-mapped shapes are. Instead, you fill them with patterns or tones that simulate a bit-mapped tone effect.

Because they allow you to work with outlines rather than bit maps that must be scaled up and down in fixed increments, vector drawing programs are usually much more flexible. You can usually zoom over a much greater range of magnifications and scale text to any size you want.

9

Converting Scanned Images

Sooner or later, you'll want to convert a scanned image from one format to another. There are many reasons you might want to do this: Your scanner controller software produces only one format, and your image editing software handles another. You have a bit-mapped image that you want to convert to vector art to simplify scaling the drawing up or down. You have access to an IBM PC at work, but you want to use some of the images on your Mac system.File and format conversion doesn't have to be confusing. All you need to know is a little bit about the various types of files that you may be working with and what software helps you convert them from one to another. You'll find that information in this chapter.

File Conversions

Your file conversion needs will break down into four categories. First, you'll often want to convert files from one bit-mapped format to another. The particular image editing software that you want to work with may not accept a file that you receive from another source.

The second type of conversion you might want to perform is to translate bit-mapped images into vector format, so they can be manipulated with a draw program such as Aldus Freehand. To do this, you usually need a vector program with an autotracing capability or a stand-alone autotrace program like Adobe Streamline.

The third type of conversion is less common for scanned images but is encountered frequently in graphics intensive work. That is, you may need to convert a vector-oriented file from one vector format to another. Obviously, you won't need to do this with a scanned image until it has been autotraced into a vector file. If you're doing that yourself, you will probably use a program that will produce the final file format you need.

However, there will still be times when you will want to convert the a vector file into another vector format, using a program that can load and save both or a stand-alone utility.

The final type of conversion is the least common and generally the easiest to do: change a file from vector back into a bit map. Nearly all draw programs let you export the image of a drawing as a bit map. It's also a simple matter to capture the image with a screen capture utility and store it in your choice of bit map file formats.

I'll address the conversions of most use to scanner owners in this chapter. But first, we'll look at the different file types. You need to learn a little about the different file types in order to understand why you need to make conversions in the first place, and how they are accomplished.

File Types

This section will describe formats used both in the Macintosh and IBM worlds, as well as several that are exclusive to IBM PCs that you should know about, in case you need to import them to your Macintosh.

In general, though, image file formats on the Mac are much simpler than in the IBM world, as most Mac applications share formats like MacPaint and TIFF, even if they have their own proprietary format. That makes exchanging files fairly easy; just use Save As... to store the file in one of the common formats.

MacPaint

The MacPaint format is the single-bit, low resolution (72 dpi) file format that formed the foundation for all paint programs that followed.

There are many MacPaint files available, both as clip art and as scanned images, that you might want to incorporate in your desktop publication. Many IBM PC programs, including Ventura, allow importing of MacPaint files.

Even though the operating systems and disk formats of IBM and Mac computers are different, you can exchange files that are saved in many standard formats. I used to do this by linking the two computers with a null modem cable and telecommunicating between them. More recently, I've purchased a Rapport disk controller, which is a tiny device that can be plugged between the Mac and a Konnect Drive 2.4, external 3.5-inch disk drive. That drive can read and write the standard Mac disk formats (400K, 800K, and 1.44Mb) as well as nonstandard 1.2Mb and 2.4Mb Macintosh formats. It also reads and writes 720K and 1.44Mb IBM formats. The Apple File Exchange program converts the files both ways. I'll describe this system in more detail later in this chapter.

Although you might think a bit map is a bit map, there is a bewildering number of ways in which to store what is essentially a collection of bytes representing the gray scale or color values used to reproduce a given pixel on a CRT screen.

TIFF

TIFF, developed by Steve Carlsen of Aldus, is an abbreviation for Tagged Image Format File. The TIFF format specifications were first developed in 1986. Each file includes collections of information, called tags that provide information on resolution, number of bits used per pixel, and many other descriptors. The basic data needed to handle a file is included in a standardized set of tags that can be interpreted by any application.

However, applications can create their own tags with information that the application needs to store with the file. A simple example of this would be a longer descriptive name or caption that is displayed when the file is loaded. Because special tags are ignored by applications that don't understand how to read them, you can exchange TIFF files between older versions of an application and newer, enhanced versions of the same software. Totally different programs can also read many TIFF files created by other applications. TIFF files can even be exchanged between Macintosh and IBM computers.

Problems arise when new types of tags include important information that must be understood by the application to reconstruct the image. For that reason, you'll sometimes find TIFF files that can't be read by other software easily.

There are various types of TIFF files, including compressed TIFF, uncompressed TIFF, and TIFF PackBits. Not all software supports all the TIFF types. The TIFF format can store both binary (1-bit) files and files with 16, 64, or 256 different tones.

Four standardized TIFF formats used by many applications are classified as B (black-and-white, or binary, information only), G (gray scale), P (palette—a number of different colors), and R (RGB—red, green, blue— color).

If your application gives you a choice, you should use uncompressed TIFF format when you know you'll be exporting files to another program, particularly a desktop publishing package. More disk space is required, but you'll avoid having to reload the file into the program that created or captured it and then save it again in the compatible format.

PICT/PICT2

These formats are so-called because they consist of a QuickDraw PICTure that can recreate an image. PICT can be used for both monochrome and color raster images, and object-oriented graphics. A PICT file can even contain a combination of both types.

The PICT2 format is a recent variation that can store color or gray scale information. Such files contain commands used with 32-Bit Color QuickDraw, and therefore can be used only with applications that run on Macs with other than a 1-bit, monochrome display.

RIFF

RIFF is an abbreviation for Raster Image File Format. Introduced with LetraSet ImageStudio, it is now shared with another product from that company, ColorStudio.

RIFF files can store gray scale information or line art with up to 16.7 million different tones. PageMaker 4.0, QuarkXpress, DesignStudio and Ready, Set, Go, can all import RIFF files.

PCX

PCX is the second most common bit map file format in the IBM PC world, but isn't found among Macintosh software. There *are* shareware utility programs, like PCxtc, that allow you to use PCX files, so I'll describe the format here.

PCX was developed by ZSoft for their line of graphics products, which includes PC Paintbrush IV Plus, Publisher's Paintbrush, and Publisher's Type Foundry. Windows 3.0 Paint is an adaptation of Zsoft's basic technology.

PCX was originally a binary file format, and gray scale capabilities were added later. As a result, while many programs can work with 1-bit PCX files, those that can handle the gray scale versions are less common. Ventura Publisher, for example, can import only PCX line art. You must convert gray scale PCX files to some other format before using them in this desktop publishing program.

For that reason alone, the TIFF format should be your first choice for storing and using gray scale images.

TGA

TGA is the TrueVision format developed for that company's Targa system. It has four modes: 8-bit Targa M-8 Color-mapped format (Type 1); 24-bit Targa 24 True Color (Type 2); 16-bit Targa 16 True Color (also Type 2); and 8-bit Targa 8 Grayscale (Type 3).

IMG

Digital Research's GEM Desktop, and programs like the GEM version of Ventura Publisher, PerForm, and ScanXpress, produce IMG files. Ventura Publisher for the Macintosh supports this monochrome and gray scale format.

Vector and Outline Formats

Outline-oriented formats are even less standardized than bit-mapped formats, even though the American National Standards Institute (ANSI) has attempted to define a common, public standard for easy interchange of vector files. Vendors have elected to implement even the standard formats in their own ways, so that even files with the same extension can't necessarily be loaded by just any program.

The fact that exchanging and converting vector files is possible at all is due to the fact that draw programs all deal with common elements, called primitives, which include lines, arcs, polygons, rectangles, circles, ellipses, and text. The difference between one program and another is the way these primitives are described. Each component may in turn have attributes, including color, line width, and orientation.

Conversion programs translate from one vector format to another by changing the descriptions of these elements as best they can. The results can vary from excellent to poor, depending on which features of the respective formats are used by an individual file. Gradient fills (or fountains) are notoriously difficult to translate from one format to another, because each program achieves them in a different way. The following are a few of the most widely used vector file formats.

GEM

Ventura Publisher users will be familiar with the GEM format, produced by GEM Draw and some other programs. This is the preferred object-oriented file format for Digital Research products and other programs that run under GEM.

Encapsulated PostScript

Encapsulated PostScript (EPS), used for storing outline images, is one of the most common of the vector file types. EPS files can be printed by PostScript printers and imported by most desktop publishing packages, including Ventura and PageMaker.

PostScript is a page description language (PDL) developed by Adobe Systems and licensed or emulated by other manufacturers. Because PostScript uses outlines for fonts and graphics, many of the benefits of vector graphics discussed earlier in this chapter apply. That is, PostScript allows great flexibility in sizing images, because a description of how to draw the image rather than a bit map is used. You don't necessarily save any disk space, however, since the PostScript language description of an image can itself be very long.

That also means that higher resolution output devices can take full advantage of their high resolution capabilities. A 300 dpi bit map printed on a 1270 dpi Linotronic imagesetter won't look any sharper. The Lino uses four of its smaller dots to simulate each of the larger 300 dpi pixels. However, with a PostScript file, the image can be printed at full resolution.

EPS files are simple ASCII files containing the program statements necessary to build an image in the printer's memory. A low resolution image header, which can be used by the applications program to display a rough approximation of the PostScript image, is appended. If your software doesn't support this function, a gray square or a box containing a large X will be substituted on-screen. You'll have to print out the page to see what the image looks like.

CGM

The Computer Graphics Metafile format is a common vector-oriented file type, available in slightly different variations for a number of different applications programs. ANSI laid out the original specifications for CGM and last updated them in 1986. Since CGM doesn't support later developments, such as Bezier curves, most vector software vendors have developed their own file formats and use CGM just as an interchange format. Some Mac applications, such as Ventura Publisher, accept this format.

WMF

Windows Metafile format may displace CGM as a common format for exchanging vector information between programs, if only because Windows 3.0 is becoming so popular, and all Windows applications support WMF. Ventura Publisher will import these files.

Other Formats:

Other bit map files you are likely to encounter include: .GIF, the Compuserve Graphics Interchange Format; .LBM, used with DeluxePaint, equivalent to Amiga IFF files; .MSP, Microsoft Windows Paint format; .PCC, used by some Zsoft PC Paintbrush products to store cutouts; .PCR the PC Rockland format used by Optiks; and .WPG, Word Perfect Graphics files.

Common vector-oriented formats include .CDR, for CorelDraw; .DRW, for MicroGrafx Designer, Freelance, and others; and .SLD, for AutoCAD Slide files. A few Mac applications, like Ventura Publisher, will import DXF (AutoCAD's Data Exchange Format).

Software Conversion

The simplest way to convert files from one format to another is to use a program that can read and write both. You simply load the image into a program using one format, then save it to disk again in a second format supported by that software.

A better bet is to use an image editing program that lets you save in either of the formats you want to convert to and from.

Most scanner software and some image manipulation packages (particularly draw packages) can save files in more than one format and thus convert between them. Included on the disk packaged with this book is a specialized conversion program, PCXtc, which will convert files from the IBM PC-based PCX format to a format your Macintosh can use. QuickGIF Plus and GrayView can also be used to convert from one file format to another.

Raster-to-Vector Conversion

If you're looking for high concept software, nothing beats the latest autotrace packages. These are designed to do one thing: trace bit-mapped images automatically. That's more challenging than you might think, because a considerable amount of processing power is required to sort a literal maze of pixels into the lines that (hopefully) make up an image.

Autotrace software works best with line art, because it has well defined boundaries of sufficient contrast for positive identification. Continuous tone images work less well. Autotrace programs tend to draw lines around each section of an image with a given tonal value. Photos that have been scanned in 4-bit mode and which display 16 levels of gray will have their posterized bands of tone dutifully outlined. Full 256-gray continuous tone images end up with hundreds or thousands of tiny objects outlined. The process can take hours and still not produce anything useful. Indeed, many autotrace programs won't handle gray scale images at all. Figure 9.2 shows a gray scale image that has been autotraced.

On the other hand, line art can be usefully translated into vector-oriented drawings. Computer assisted drafting, in particular, depends in large part on the ability of a program to scan in existing drawings for update or adaptation. It would simply take too long to redraw a detailed plan, even using the faster techniques possible with CAD. (Many firms do this out of necessity, ignoring the cost in productivity.) Most engineering organizations have vast files of drawings that they either digitize using tablets or convert to outlines with autotrace software.

For example, printed circuit board designs can be scanned from prints made of the film masters. Enlarged prints can provide the necessary detail to allow a good scan. The exact shape of the traces and pads isn't as critical as the distance between them, so autotracing can provide drawings that can be efficiently cleaned up, modified as required, and output to negatives to expose new boards.

More critical are other engineering and architectural drawings that are being converted to outlines. In desktop publishing, we also use autotracing to capture outline art that will be enlarged or combined with other draw-type graphics.

Autotrace systems use mathematical algorithms to determine the boundary between one portion of an image and the next. This boundary is then traced to form a line. Vector-oriented software always uses mathematical formulas to represent such lines; that is why you can enlarge and reduce objects without losing resolution. So, the edges of either polygons or Bezier curves are used to describe the complex lines that make up a typical scanned image. Some programs use both polygons and Bezier curves.

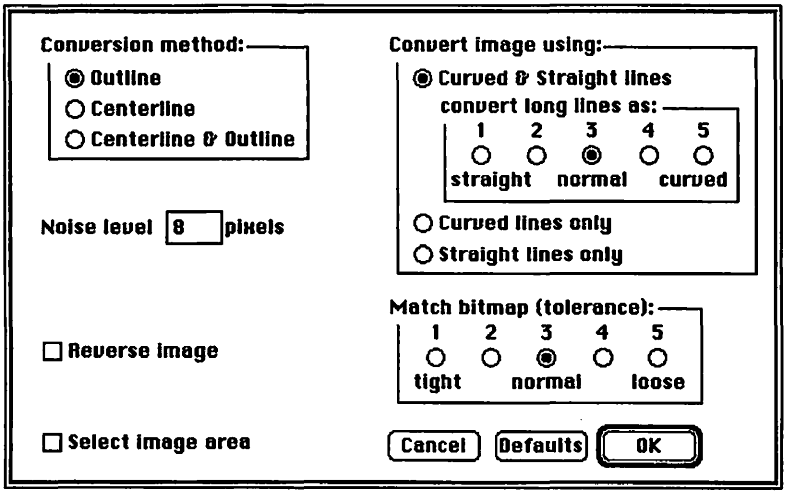
The number of points between line segments determines how accurate your traced image will be. The more points used, the smaller the image element that can be represented and the more closely the object resembles the traced image. Of course, a large number of lines makes your object as jaggy as the original image.

Most autotrace programs allow you to adjust the accuracy or tolerance used to trace the image. If you specify high accuracy—a small tolerance for variations—many short lines are used to trace the image, producing a somewhat jaggy-looking image. If you ask for reduced accuracy (a higher tolerance for variations), the program uses fewer lines and smooths out the jaggies. You can also use your vector image editing software to adjust the number of control points on the lines that define an object.

Depending on how skillful the programmers were, autotrace software can range from slow and accurate to fast and sloppy. A rare few are fast and accurate. The autotrace functions built into draw programs are generally of acceptable quality, while the stand-alone programs, which include Adobe Streamline are usually a step ahead in speed and quality. Streamline is shown in Figure 9.1.

As noted, autotracing works best with images that have distinct edges. The best have solid forms or silhouettes as the key image areas. In that respect, scanned line art (like most logos) is often ideal. You can convert the bit-mapped lines of the scanned original to a vector-oriented

Figure 9.1



Adobe Streamline was one of the first autotrace programs for the Macintosh, and was the first standalone program for this function.

drawing that can be touched up, manipulated, enlarged, or reduced as you wish. Conversely, photographs are usually poor subjects for autotracing unless they are very high in contrast.

Most autotrace packages allow you to choose contour tracing or line tracing. When you ask for line tracing (also called centerline), the vectorizing is performed in two steps. First, the lines of the existing drawing are thinned. That is, the lines that make up the image are eroded until they are only a single pixel thick. This works well if an image is, in fact, made up of various sized lines. However, if other oddly shaped solid areas are used to make up the image, you may get an image that doesn't resemble your original very much.

After thinning is completed, the program works the same as it would with contour tracing. The edges are replaced with line objects that reproduce the image.

Exchanging Between Platforms

Today, the concept of exchanging scanned image files between platforms has several meanings. Some of the high end page makeup and color separation systems can accept images scanned on modest IBM PC-compatible systems. It's also possible to exchange scanned images with Unix-based systems running X-Windows or some other shell, such as the Next operating system.

For most of us, though, the two key platforms are the IBM PC and the Macintosh. You may want to get images that have been scanned or manipulated by an IBM PC. That's not unreasonable. My Epson color scanner is attached to both a PC and a Mac, but when I want a color image for the IBM PC, I still usually scan it on the Mac, do most of the work, and then transfer it over to the PC. I don't have a 24-bit color display system for my PC, but I do for my Macintosh II.

The good news is that you can quite easily transfer files from the Mac to the PC. Getting images you've scanned on a PC into a Mac is only a little more difficult.

While I was writing this chapter, I scanned a single image on the Mac and saved it in a variety of formats. These included the TIFF and EPS formats common to both Pcs and Macs, and Apple's own PICT and MacPaint formats. I transferred them over to the PC and loaded all of them into various PC software packages that accepted those file formats. TIFF was easiest to exchange, of course, but even the low resolution MacPaint files translated with no problems.

I also transferred some images from the PC back to the Macintosh. There, I had the most luck with TIFF and EPS files created on the PC. That really wasn't much of a problem at all. Later in this chapter I'll detail some techniques that can make file exchange almost transparent.

Macintosh to PC

There are several ways to transfer image files from a Mac to a PC. Any Mac with a Macintosh FDHD (1.44Mb) drive can exchange files. While the formats used by the Apple Hierarchical File System (HFS) are different from those used by the IBM PC, the Mac FDHD can read both. You'll need a Macintosh utility program called Apple File Exchange, which is furnished with your System software, to translate the files from one to the other.

Just load AFE and insert a disk formatted for the IBM in the Mac's FDHD drive. A window like that shown in Figure 9.x is displayed. You then specify the file(s) to be translated on the Mac side of the window and click on Translate to copy them to the IBM DOS side of the window. The process takes only a few minutes.

You can then take the DOS disk to your PC, copy the files, and load them normally. Any translation or reordering of the bytes (the Motorola microprocessor found in the Mac and our own Intel chips use a different arrangement) is done automatically.

If the Mac doesn't have an FDHD drive, there are other options. Dayna Communications offers software called DOS Mounter that can be used with Dayna's own hardware or compatible hardware to allow the Mac to read and write DOS disks. I use a gadget from Konnect Technologies called Rapport, which can be used with anything from a Mac Plus on up to allow reading DOS disks.

With a Mac Plus and 800K Mac drives, you can read only 720K IBM disks. Rapport can also be used with Konnect's own Drive 2.4 with a Mac Plus or later Macintosh model, including earlier Macintosh II systems that didn't have 1.44Mb FDHDs.

That's the solution I selected for my own Mac II. The Drive 2.4 can read and write 720K and 1.44Mb IBM disks, and it reads and writes standard 400K, 800K, and 1.44Mb Macintosh formats. In addition, it allows formatting 800K Mac disks for 1.2Mb of storage and 1.44Mb disks for 2.88Mb. I have added a utility program called Access PC, which is an INIT that dispenses with the need to use Apple File Exchange at all. DOS disks appear directly on my desktop; files have icons associated with them, and DOS subdirectories appear as folders. This under-$100 program is essential for any one who has a compatible floppy drive and needs to exchange files with the PC world regularly.

The recent introduction of new Macintosh systems like the Macintosh Classic, Macintosh LC, and Macintosh Si have strengthened the presence of this platform in the business environment. Macs won't go away, and Windows 3.0 won't replace them. So, it behooves us as Macintosh users to learn how to co-exist with or, better yet, take advantage of the PC's capabilities. Learning to exchange scanner files is an important first step.

You can also get PC drives to read Macintosh disks if you use a Copy II PC Option Board from Central Point. This board was originally designed to defeat bothersome copy protection schemes by giving you greater control over your disk drive functions, but as a byproduct it can also enable your drive to read and write Macintosh disk formats.

PC to Macintosh

You'd think that the methods and equipment described above would be sufficient to allow going the other way—from the PC back to the Mac. Depending on the software you use for the conversion and the file format, that's not always the case. This next section is aimed at readers who use both Pcs and Macintoshes regularly or who work with someone who is familiar with the Mac. Some of the terminology and procedures may seem a bit foreign to Macintosh neophytes.

You usually tell your IBM PC what type of file it is going to load ahead of time. PC applications generally have no way of telling what type a file is. Instead, you check off a button or other indicator before attempting to load the file.

The software then checks the file format to determine if it is compatible. Sometimes it isn't; if you try to load a compressed TIFF file into a program that accepts only non-compressed versions of TIFF, your application will report an error and abort the loading process.

Macintosh files are a little different. Each file has a header that tells both what type of file it is (TIFF, EPS, MacPaint, and so forth) and the application that created it. The latter code makes it possible to launch Macintosh applications by clicking on a file that has been created by the application. The Mac operating system examines the file, determines the appropriate application needed to load it, finds the application, and then loads it along with the file you selected.

Unfortunately, when you copy a file from an MS-DOS disk using Apple File Exchange or some other utility, it isn't always possible to tag the converted file properly. Your utility doesn't necessarily know the name of an appropriate creator application nor the type of file you're converting. You can't tell a Mac application that it will be loading a TIFF file, then select the file name. Mac programs don't even recognize the existence of files that aren't properly tagged with a compatible file type.

The creator application may not matter: TIFF files may be labeled DIDR if they are created by Digital Darkroom, PHOP if created by Photo Press, or ST32 if created by Studio/32. Any of these programs will load a TIFF file created by any of the others.

Your TIFF files transferred from the PC may have neither creator nor file type attributes specified correctly. Apple File Exchange will label them as bina (binary) files if it doesn't recognize them as ASCII, RFT, or another recognized format. The creator type will be tagged as MS-DOS. You can view the attributes for any file using a utility like RESEDIT and change them to something more appropriate if you wish.

It's always best to work with a copy of the original file if you haven't used RESEDIT before. However, making the changes isn't very difficult. Simply launch RESEDIT. Select the file you want to modify, and press Command I (or the Get Info choice from the Edit menu). You'll be shown a dialog box. For TIFF files, type in TIFF as the file type, and enter the code for the TIFF-compatible Macintosh application you have on that machine. In addition to telling the Mac which application to look for when you double-click on that icon, the code also tells the system which style icon to use.

If you don't know the code for your application, open a TIFF file created by that application and see what creator code is used.

Once you've done that, your application should recognize the file as one of its own.

Three More Sophisticated Solutions

Access PC and MacLink Plus and SoftPC all eliminate the need to use RESEDIT. These are three DOS-to-Macintosh (and back again) solutions which take a comprehensive approach to solving the file interchange problem. I have all of them, and use them in slightly different ways. Each makes the conversion process almost transparent. I'll describe each in turn so you can decide for yourself which is best for you.

MacLink Plus/PC

This package is actually a rich collection of software for your Macintosh and PC, and a cable you can use to connect your Mac to a PC, another Macintosh, or even a non-DOS workstation such as Sun Microsystems or NeXT machines.

MacLink Plus' strength are its translators. More than 150 different file translators are available which allow you to convert to and from a broad range of scanner and vector image file formats, and formats used by other applications, including word processing programs and spreadsheets.

The Macintosh portion of MacLink Plus includes DOS Mounter, which will allow your FDHD SuperDrive to read DOS disks directly. You can see these disks and their files as icons and folders on your desktop; there is no need to use Apple File Exchange as an intermediary.

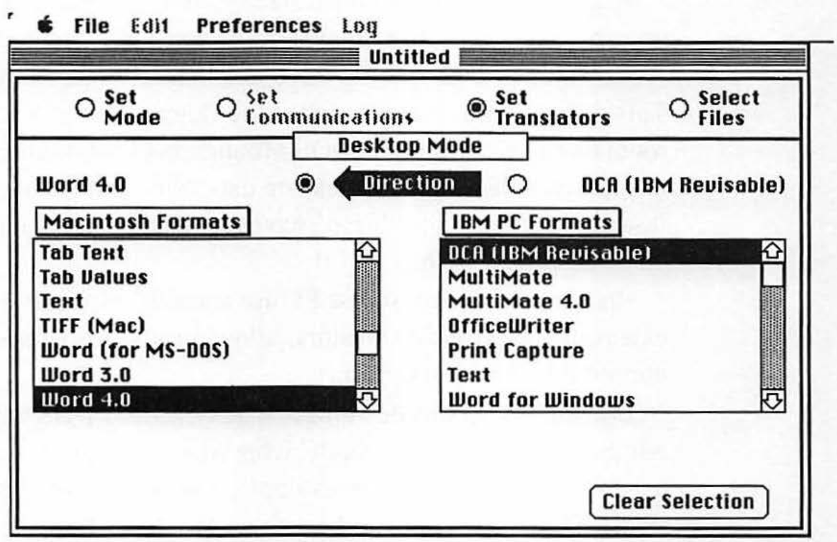
The main MacLink application has several modes. In Desktop mode, it will use any of the translators to convert files you have already brought over to your Macintosh – usually by a direct disk swap. DOS Mounter makes it easy to use MS-DOS disks in the Mac; then you use MacLink to convert the files.

MacLink can also be used to transfer files from the PC or another Mac over a modem connection (using telephone lines) or by a direct cable connection. If the two computers are close enough, you can link them with the provided cable.

In either case, both the PC and Macintosh need to run their own versions of the MacLink software. Either computer can function as the master or slave system. For example, when the PC has its MLPC.EXE program loaded, the Macintosh can connect directly (through the special cable, which is configured as a null modem). Or the Mac can dial up the PC and "talk" over longer distances through a pair of modems.

In this mode, the Macintosh views its own drives and those of the PC through a pair of scrolling windows. You can select files on either machine and send them to any folder or subdirectory on the other that you choose. You also specify the appropriate translator combination (say PC TIFF to Macintosh TIFF). The transfer takes place at up to 56K bps, with translation done automatically. MacLink is shown in Figure 9.2.

Figure 9.2



MacLink Plus

You can also put the Macintosh into so-called host mode, and load the MacLink CONSOLE.EXE package on the PC which will allow the PC to take control of the transfers.

DataViz, the folks who market MacLink, have another solution. They are now bundling Dayna's DOS Mounter software with their MacLinkPlus 5.0 software. This combination provides a way for Macintosh FDHD SuperDrive disks to recognize DOS disks, plus conversion utilities to translate files from one format to the other.

The MacLinkPlus translators handle some 250 combinations of word processing, spreadsheet, graphics and database files. There is support for Word for Windows, WordPerfect 2.0, FrameMaker, and Windows graphics. You don't even need a FDHD drive, since DataViz provides you with cables to connect the two computers directly, if you like.

Access PC

Access PC is another utility that lets you see DOS disks as icons on your Mac desktop. It has some advantages of its own. For example, DOS Mounter must write some Desktop information to your PC disk. This doesn't harm the disk, but if the floppy is so full that there is no room for the Desktop data, DOS Mounter may not be able to open your disk. Access PC *can* write Desktop data, which allows it to open a disk faster the next time, but doesn't have to. In fact, you can turn this feature on or off as you wish.

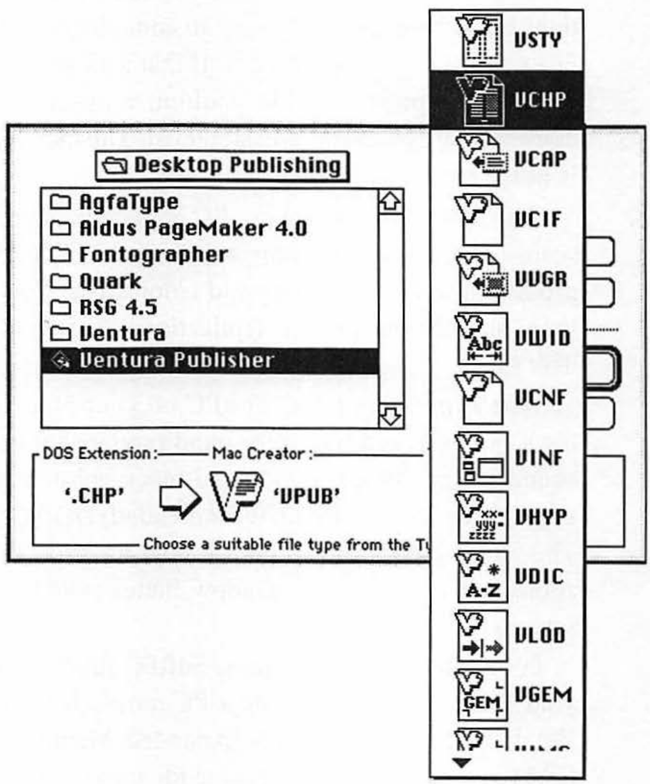
Both MacLink and Access PC use a handy system of linking PC file extensions to Mac file creators, allowing you to bypass the ResEdit nonsense I discussed earlier.

Instead, you associate various type types and DOS extensions, and tell the MacLink or Access PC what Mac application and file type to use for that kind of file. For example, you can tell them that when they find a file with a .TIF extension on an MS-DOS disk, you'd like to tag that disk with the TIFF file type, and a creator that matches a TIFF-capable program you own.

This is even easier to do than you might expect. Access PC will allow you to browse among your current applications, displaying the various icons and file types that the particular application can handle. When you find the one you want, you can choose it by clicking on the icon.

Once you've told Access PC how to handle the various MS-DOS file types, everything is automatic. When you open an MS-DOS disk containing TIFF files, they'll be represented by the appropriate ICON, just as if they were created by a Mac application. Clicking on one of these icons will launch your application and load the file, just as it would with a "native" Macintosh file. Access PC is shown in Figure 9.3.

Figure 9.3



Access PC will associate your applications' icons with the file extensions of IBM PC files, and display those icons automatically.

SoftPC

This amazing software is, in a way, the ultimate Mac/PC file exchange solution. With SoftPC, your files never have to leave your Macintosh to move from Mac applications to PC applications and back. What SoftPC does is take a portion of your Mac's resources, including your hard disk drive, and construct a little working PC inside. You can choose an 8088-compatible PC-XT, or a more sophisticated 80286-based PC-AT with math coprocessor support and EGA graphics!

Since you're used to a speedy Macintosh, you'll find these emulations range from painfully slow to annoyingly slow, depending on how fast your Mac and hard disk is. If that's all you look at, though, you're missing the point. It's like faulting a chess-playing dog because it doesn't play at the Grandmaster level. The fact that SoftPC works at all is little short of astounding.

SoftPC lets you run many PC applications on your Mac, including scanner image editing software, if you have the EGA module that can produce a selection of gray and color tones. You can go directly from those applications to Mac applications, sharing some of the same hard disk space.

When you've installed SoftPC on your Mac, it will "boot up" with the same beep and whirring sound produced by a real PC-AT. It's just sound effects, however. Then, a black-colored window will open on your screen, and the familiar (or dreaded) DOS C:> prompt will appear. The image is sized to fit on a 9-inch monitor, but you can click on a zoom box to produce a window better suited for 13-inch and larger screens.

For all intents and purposes, SoftPC functions just like a real PC on your Mac. You can emulate a PC mouse if your software requires it. What the IBM world calls Expanded Memory Specification (EMS) RAM is available (you can set aside up to 4 megabytes, depending on how much RAM your Mac has). There is even support for the Intel 80287 math co-processor.

I installed a brace of PC applications I have, ranging from Microsoft Word 5.5 to Publisher's Paintbrush, with varying degrees of success.

Those two happened to work just fine. I actually got a lot farther installing Microsoft Windows 3.0 on my Macintosh than I got with several 100 percent PC compatibles, but my Mac finally rebelled, and I haven't tried *that* again. Insignia, the vendor of this software, claims Windows support. Since I needed to tweak Windows quite a bit to get it running on several PCs, I suppose there are some things I need to do to get SoftPC to run it, as well.

If you have a 1.44Mb Superdrive, or a compatible drive like my Konnect Drive 2.4, you can tell SoftPC to use that as your Drive A: (which is what DOS calls its first floppy drive.)

A hard disk called Drive C: is created for you automatically. This one is an exclusive resource of SoftPC and can't be accessed, by your Macintosh applications, unless you also have AccessPC, in which case your Mac can treat these volumes as just another Mac folder.

A second drive, Drive E:, can be created to represent any file, folder, or even hard disk in your Mac. This Drive E: gives SoftPC access to any of your Mac files. Your Mac applications, in turn, can work with any files saved in the Drive E: folder by SoftPC. A third hard disk, Drive D: can be created as another SoftPC exclusive drive. You might want to do that, since Drive C: has limited room, if you want to keep some additional files and applications separate from your main Macintosh volumes.

SoftPC also includes a "slave" program you can run on a PC which will, when the PC is connected to the Mac with an appropriate cable, allow SoftPC to use the PC's Drive A: as it's own floppy. You might want to do that if you don't have a Superdrive or compatible, or need to transfer some files that are only available in 5.25-inch format.

Summary

Converting files is something all scanner users must face at one time or another. Your scanner controller software produces one format, and your image editing software handles another. You have a bit-mapped image that you want to convert to vector art to simplify scaling the drawing up or down. You have access to a Macintosh at work, but you want to use some of the images on a PC system. You'd like to convert a scanned image into a font that can be printed with your LaserJet III printer.

File and format conversion doesn't need to be confusing. But you do need to understand the various types of file formats.

PICT and PICT2 are two very flexible native Macintosh formats that can store bit-mapped images, object-oriented graphics, or a combination of the two in a single file.

TIFF is an abbreviation for Tagged Image Format File. This most common of image file formats has four different types: B (black-and-white, or binary, information only), G (gray scale), P (palette – a number of different colors), and R (RGB – red, green, blue – color). These classes can be compressed or uncompressed.

RIFF is a bit-mapped image file format used by many LetraSet applications, and supported by other packages.

PCX is the second most common file format in the PC world. It was developed by Zsoft for their line of graphics products. PCX was originally a binary file format; gray scale and color capabilities were added later.

Encapsulated PostScript (EPS) is the most common of the vector file types. EPS files can be printed by PostScript printers and imported by most desktop publishing packages, including Ventura and PageMaker.

TGA is the TrueVision format developed for that company's Targa system. It has four modes: 8-bit Targa M-8 Color-mapped format (Type 1); 24-bit Targa 24 True Color (Type 2); 16-bit Targa 16 True Color (also Type 2); and 8-bit Targa 8 Grayscale (Type 3).

The MacPaint format is used with the Apple Macintosh and is the single-bit, low resolution (72 dpi) file format that formed the foundation for all paint type drawing programs that followed.

IMG and GEM are supported by Ventura and other programs that use the GEM Desktop, including ScanXpress. These are bit-mapped and vector formats, respectively.

Other important file formats include AutoCAD's DXF, Hewlett-Packard Graphics Language (HPGL), the Compuserve Graphics Interchange Format (GIF), and Computer Graphics Metafile (CGM).

The simplest way to convert files from one of these formats to another is to use a program that can read and write both. You simply load the image into a program using one format, then save it to disk again in a second format supported by that software.

Autotrace systems use mathematical algorithms to convert bit-mapped art to vector files. The number of points between line segments determines how accurate your traced image is. The more points used, the smaller the image element that can be represented and the more closely the object resembles the traced image. Most autotrace programs allow you to adjust the accuracy or tolerance used to trace the image. If you specify high accuracy—a small tolerance for variations—many short lines are used to trace the image, producing somewhat jaggy outline.

Depending on how skillful the programmers were, autotrace software can range from slow and accurate to fast and sloppy. A rare few are fast and accurate. Autotracing works best with images that have distinct edges.

More of us are finding a need to exchange scanned images between IBM Pcs and Macintosh computers. Any Mac with a Macintosh FDHD drive (a 1.44Mb drive) can exchange files. While the formats used by the Apple Hierarchical File System (HFS) and IBM PC are different, the Mac FDHD can read both. You'll need a Macintosh utility program called the Apple File Exchange to translate the files from one to the other, though. Moving files from the PC to the Mac sometimes requires a little extra work to get the Mac to recognize the foreign file formats. MacLink and Access PC are two software add-ons for your Mac that can automate this work for you. SoftPC goes one step further by actually emulating a working IBM PC in your Mac.

10

Optical Character Recognition

Of all the things you can do with your scanner, optical character recognition (OCR) may have the broadest applications, the least understanding among users, and the greatest diversity in software quality. Nearly everyone has a use for OCR at one time or another, but nobody really knows how to use it effectively. You can pay $500 for a great OCR package or $1000 for a poor one. With all these contradictions, it's no wonder that OCR is destined to be one of the boom technologies of the 1990's. There may even be a place for consultants who can explain OCR to businesses before they lay out big bucks for a solution that may not work.

The reason is simple: the arrival of the paperless office will probably coincide with the arrival of the paperless bathroom. Some 95% of the information we work with is still stored on paper. Even as we approach a computer density of one per desktop, business continues to thrive on paper output. You can read paper reports anywhere, shuffle through stacks of them on your desk, make copies of them, and do lots of things that even the best desktop metaphor won't handle. If anything, personal computers have created more paper in the form of desktop publications and reports. High speed laser printers make it as easy to generate 100 copies as five.

Paper remains our most important source of information and our most popular way to exchange it with others. Yet, hardcopy data is of absolutely no use to a computer. The information must be captured and converted to the numeric codes that the computer can handle. That's where OCR comes in. Optical character recognition isn't a way of eliminating paper; it just helps you live with it.

OCR software faces some of the stiffest demands this side of 3-D modeling. It must be able to recognize a tremendous array of type styles and sizes. Sometimes characters touch. Is the ligature *fi* two characters, *f* and *i*, or is it an *h*? Figure 10.1 shows an example of the wide variations in type that an OCR program is expected to discern.

Of course, the page an OCR program is scanning may contain a lot more than simple text. There may be graphics that the program has to ignore or, perhaps, capture and dump in a separate file. Text may be arranged in columns. Good OCR programs recognize those columns

Figure 10.1

OCR software can read
monospaced type easily.

Calligraphic fonts cause
serious problems.

Forget about capturing most
foreign alphabets and odd styles.

Ordinary proportional fonts can
be read with near 100 % accuracy
by high-end OCR software.

OCR programs are expected to be able to read all these kinds of type, including monospaced typewriter type, proproportional fonts, oddball typefaces, characters which touch one another, faded type, and special characters.

and retain the format if you wish. They are smart enough to insert the correct formatting codes in the word processor of your choice. Some can tell boldfaced text from ordinary text and even recognize a few different typestyles. One group of OCR software can be trained to recognize the most exotic typestyles, although this is a tedious procedure. Another can read almost any typestyle with no training at all—a minor miracle considering the variations among fonts.

This chapter explores some of the key features of OCR software. You'll learn how OCR works and what you need to know to work with it.

Types of OCR Products

Three is the magic number this chapter. In it you will find three kinds of OCR products, three ways to implement them, and three methods of recognizing text. First, we'll look at the trio of OCR classes.

Intelligent scanners, or compound document processors, are hybrids that can differentiate between graphics areas and text in a single document. These are expensive devices and are not generally used for desktop publishing. Within areas containing graphics, such a scanner functions as a bit-mapped graphics scanner, capturing the image-intensive information. It also recognizes text information and attempts to interpret the alphanumeric characters in a form the computer can use.

Page scanners are capable of reading text information anywhere on a page, but they ignore graphics. This type of scanner is useful when many different kinds of documents—incoming correspondence, for example—must be scanned and interpreted. A forms reader is the third, more specialized type of OCR, which can be trained to read specific document layouts, scanning for typewritten, printed, or handwritten information. A forms reader is generally faster than a page scanner, because it ignores any incidental text that may appear on a page in favor of the key information.

Ways to Implement OCR

Compound document scanning, text-only OCR, and forms reading can all be found on three different types of platforms. At the high end are dedicated OCR systems that do nothing but scan documents and convert the required information into a form the computer can handle. Such equipment can be used for compound documents, text only, or forms, depending on the application. Insurance companies, for example, may use a dedicated OCR to capture handwritten applications, reducing the amount of keying that must be done. Heavy duty word processing departments may have text-only OCR machines to capture typewritten or printed material for computer processing.

Finally, there are software-only solutions for OCR applications. You can find OCR software for a few hundred dollars that works on any PC compatible, accepts data from hand scanners, flatbeds, or sheetfed machines, and operates acceptably, if slowly. The evaluation copy of PRO-CR included with this book can be registered for even less.

High end software demands a fast 68020 or 68030 Macintosh, lots of memory, and, even at that level, some patience. Twenty seconds may not sound like a lot of time to read a page, but when you multiply it by 1000 pages, you can look forward to a long wait.

If you are doing a fair amount of OCR work, you'll certainly want to consider one of the high end packages that run on a faster Mac computer and an automatic document feeder for your scanner. You may not care how long the process takes if you don't have to sit there watching it.

Text Recognition Methods

An OCR program uses three basic methods of recognizing text. It can compare the characters it encounters with a set of patterns until it finds a match. It can also look at the individual elements that make up a character, deciding that two diagonal lines and a crossbar must be an uppercase A, regardless of how they are arranged. OCR software may also take a hybrid approach, using either or both techniques coupled with some artificial intelligence, spelling dictionaries, or other technologies. I'll discuss each of these in turn.

Pattern Matching

The simplest OCR programs use pattern matching, also called matrix matching. This method is very flexible simply because you can train the program to read new typefaces just by providing a new set of patterns to match against.

Pattern matching software divides each character into a matrix of blocks. The sample characters in its library are divided into a similar matrix. The software then compares the two to find a match. The number of congruent bits helps the software decide whether it has found a match. The system works because the number of possible matches for any given character is limited—typically 100 or so upper- and lowercase letters, symbols, and punctuation. If a character scores significantly higher against one sample than another, odds are very good that it is, indeed, that character.

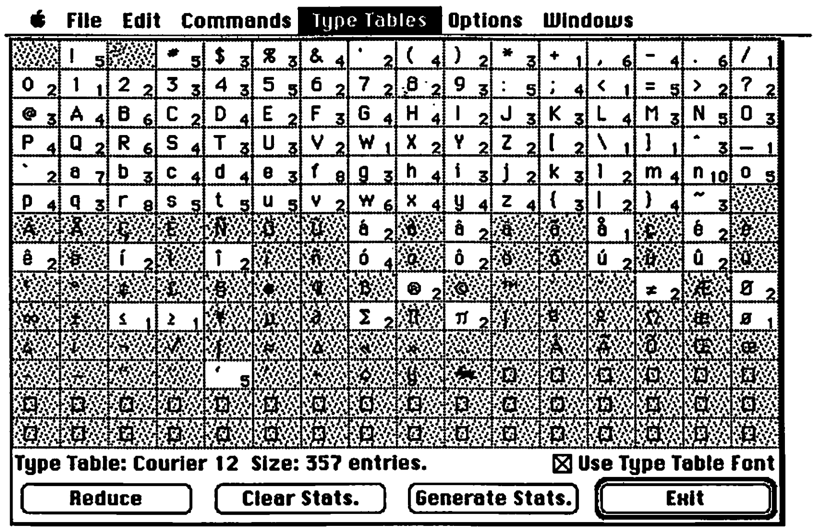
The odds improve when you select the typeface for the OCR software, telling it to use its Courier, Elite, or Times Roman font for comparison. You may even be able to specify whether the font is proportionally spaced or monospaced. Given that much information, the software uses its best character library and can score very high in recognizing your text.

Accuracy drops considerably when the OCR program must attempt to guess which pattern library to use and when it is trying to read a new text font that matches the existing libraries poorly.

Figure 10.2 shows a character library used by a popular pattern recognition OCR program.

Pattern matching is a fairly flexible scheme. Scanned characters don't have to match the ideal patterns exactly; they need only score high enough to allow the software to make a good guess. The most sophisticated packages allow you to specify how much variation is allowed, so you can fine tune the system for high quality, low quality, and damaged originals.

Figure 10.2



Pattern matching allows OCR software to compare the general outline of characters it is attempting to recognize with a library of templates.

Some software makes it possible to check against several pattern libraries simultaneously, which is useful for fonts that don't quite match a specific typestyle and for pages that contain mixed typestyles.

Pattern matching software can also be trained to recognize new typefaces. This is a tedious procedure that involves scanning a sample of the type style and then identifying each character individually. The process speeds up as the software gradually begins to recognize characters. You usually have to add several variations for each character to a given library to account for letters that aren't 100% uniform. In a few cases, you'll manage to fill up the available spaces in the library and be unable to add any new variations.

Feature Extraction

Topological feature extraction is another technique for interpreting characters on a page. Each symbol is broken down by the features that make up that symbol. That is, a letter A consists of a crossbar and two slanted lines joined at the top. As the OCR operates, built-in algorithms extract the features of the character being read and provide a match. The chance of spotting an A, no matter what font happens to be used, are excellent. Feature extraction can even be used to interpret constrained handwriting — that is, characters printed with a modicum of care roughly on or inside a box printed on a form (in an ink invisible to the OCR).

Because no character has more than about 20 different features and because a long list of patterns does not have to be compared, feature extraction has the potential to be a very fast OCR method. OmniPage is an example of this type of software.

Hybrid Systems

Many OCR packages use a mixture of techniques to increase their accuracy. They may rely on pattern matching or feature extraction to narrow their choices and then use other algorithms to make a final selection.

Context checking is one method that is used to break ties. The characters that appear on either side of a questionable character can provide important information. A lowercase l or uppercase O found in the middle of a group of numbers is probably a numeral. The characters 5t. are probably the abbreviation St., while 8old has a good chance of being Bold, particularly if it appears at the beginning of a sentence. As you can see, some artificial intelligence techniques can be used to identify characters with greater certainty.

Some OCR programs have word lists and dictionaries. If no word can be found that matches a given set of characters, the OCR program can substitute characters for the most questionable ones in the group until a valid word is found. For example, since the letters a and s are frequently confused by OCR software, a check of a word list would tell a smart program that "becsuae" might be "because."

Statistical modeling can also be used to improve guesses. That is, how often does the letter combination that would result from a given choice actually occur in English? An h follows a t a high percentage of the time; a k does not. So the software could easily differentiate between the somewhat similar h and k. It might even be useful (someday) to have OCR software that could build a statistical model for a specific document. That would help improve accuracy even when reading highly technical material containing jargon not found in standard dictionaries.

OCR and Resolution

With both pattern matching and feature extraction, the higher the resolution of the image, the greater the potential for accuracy. Resolution is less critical in feature extraction systems, because the software is trying to identify only strokes; it doesn't much care how they are shaped.

With pattern matching systems, higher resolutions offer more information for the comparison, but at a cost in speed. Increasing the resolution four times from 75 dpi to 300 dpi provides 16 times as much information (5625 versus 90,000 bits per square inch), so your software takes 16 times as long to work.

Indeed, some OCR software limits the typesizes it can work with at various resolutions for exactly this reason. After all, you don't need 90,000 bits of data to identify 72-point type; you can get by with less. Higher resolutions are most useful for very small type. So, some packages permit reading text from 4 to 12 points at 400 dpi, 6 to 16 points at 300 dpi, and 8 to 24 points at 200 dpi.

The 200 dpi setting embraces just about every type size you're likely to want to read and therefore makes the best compromise for most applications.

A package rated at 60 characters per second on a 386 computer at 300 dpi works 20 to 25% faster at 200 dpi, because there is less information to process.

If anything, you should be more concerned about the quality of your scanned images. Make sure the glass platen on a flatbed scanner is clean and that your original is of the best quality. A tiny defect that you can ignore or retouch on a graphic image can cause an error that must be found and fixed in a text file.

Using OCR with Different Document Formats

In desktop publishing, documents that you want to capture with OCR generally have no set format. Some appear as a single column of typewritten text; others consist of several columns laid out newsletter or newspaper style. Most OCR packages can handle these varying formats easily—either automatically or by asking you to define manually which zones are to be read and in which order. In the latter case, you are presented with a reduced image of the page and allowed to place boxes around the zones you want to define.

It is possible that you will be reading the same document format month after month. For example, you receive a publication from your company's overseas office intended for distribution there and you want to capture certain sections for manipulation and merging with your own newsletter. Several OCR packages let you define page layouts with their own zones, which you can recall at any time.

The newsletter example is not the most likely scenario. A better one is a company that wishes to design forms with certain information placed in standardized locations. This data can include the name of a customer, account holder, or insured; a Social Security number or other identifying number; and other key information. Such forms could be quickly scanned by an OCR package using a predefined form layout.

This type of system would be particularly effective in applications such as waybills, invoices, and so forth, where not all documents are retrieved, . Instead of taking the time to scan the full text—which may or may not be needed—you can capture only the indexing data and rely on low cost, high density microfilm for permanent storage.

Even more intelligent OCR devices—devices that can interpret entire pages and differentiate between text and graphics—will also grow in importance. Such machines may learn to recognize many different types of forms and documents simply from the information they contain. Today, preprinted form numbers, bar codes, and other techniques must be used for this.

OCR Problem Areas

You should know in advance that no OCR product reads text 100% error free. Some of them come close, however—especially when compared to the products available only a few years ago. Good quality original text can make all the difference.

I recently received a one-page, single-spaced news release that I needed to modify. I didn't look forward to retyping it, so I slapped it on the bed of my ScanJet and let OmniPage work on it. There were exactly two errors in the whole document, so a 30-minute typing job was transformed into a five-minute scanning exercise.

At the other end of the spectrum, I used OCR extensively in preparing the list of manufacturers in Appendix D of this book. I collected a large stack of lists of addresses, scanned them, cleaned up the misspellings, and then merged and sorted the listings. These lists weren't clean copy in Courier typefaces. They were printed on colored paper, in tiny type, and in mixed typestyles. OmniPage had quite a few problems with boldfaced text that touched, confused many of the tiny letter a's (I could have done a global search to correct all the Son Joses and Sonto Cloros I found). But the hard parts to correct, the numbers, came through clear and crisp. Though I spent a few hours fixing the listings, I still saved a lot of time.

Knowing just what elements of a page can give your OCR program fits can help you avoid the jobs that won't save you much time and, in some cases, make some corrections ahead of time to make things go smoother. Listed here are the common OCR trouble spots.

- **Underlined text.** Underlining often causes descenders to run into the line, which can confuse an OCR program.
- **Joined text.** Ligatures, found in typeset text, join two characters together to make them more readable to the human eye. They can cause

problems for your OCR software, however. Trainable OCR packages allow you to specify ligatures as unique characters, so they may have less problem with them than omnifont packages. Poor quality printing can also join characters in unfamiliar ways.

- **Broken text.** If your text is fractured, your OCR software may be unable to interpret a character or may see it as two characters. If the lowercase m is broken at the end of the first hump, it may be interpreted as an r followed by an n.
- **Kerned text.** Tightly spaced material can usually be handled by software that uses feature extraction, because it doesn't care how closely characters are spaced as long as they don't touch. With pattern matching software, part of one character intrudes into the matrix of the next, confusing the program.
- **Boldfaced characters.** Doubling the line thickness of characters can cause them to run together, producing joined text that is difficult to read.
- **Italics.** Slanted letters can distort the features of a character enough to confuse an omnifont program and join characters together enough to cause problems with both types of software.
- **Photocopied text.** The inevitable quality losses that come with photocopying text can introduce faded areas, broken characters, and poor contrast. You may be able to adjust the photocopier's contrast to produce a better, more readable copy.
- **Dirt.** Any noise on the page causes problems with any OCR program. Dirty patches may look like characters to your software. If they touch a character, they can change an O into a Q or an F into an E. Use the cleanest original you can.

Summary

Optical character recognition (OCR) has some of the broadest applications of any scanner technology, but it is poorly understood. The quality of the available software also varies quite widely. This relatively immature technology will become one of the boom areas of the decade.

Text being scanned may contain both text and graphics, as well as formatting information. To capture this data, vendors have produced three kinds of OCR products, three ways to implement them, and three different methods for recognizing text.

Intelligent scanners, or compound document processors, are hybrids that can differentiate between graphics areas and text in a single document. Page scanners are capable of reading text information anywhere on a page, but they ignore graphics. A forms reader is trained to read specific document layouts, scanning for typewritten, printed, or handwritten information.

Compound document scanning, text-only OCR, and forms reading can all be found on three types of platforms. At the high end are dedicated OCR systems that do nothing but scan documents and convert the required information into a form the computer can handle. Pcs equipped with specialized high speed OCR coprocessor boards provide similar power in a package that may cost only $2500 or so. A coprocessor system may provide the only way to do the most sophisticated OCR work on a slower computer.

There are software-only solutions for OCR applications, costing a few hundred dollars to $1000 or more. High end software demands a fast Mac computer and lots of memory.

There are three basic techniques an OCR program can use to recognize text. It can compare a given character with a set of patterns until it finds a match. It may also look at the individual elements that make up a character, deciding that two diagonal lines and a crossbar must be an uppercase A, regardless of how they are arranged. OCR software may also take a hybrid approach, using either or both techniques coupled with some artificial intelligence, spelling dictionaries, or other technologies.

With both pattern matching and feature extraction, the higher the resolution of the image, the greater the potential for accuracy. Resolution is less critical with feature extraction systems, because the software is trying to identify only strokes; it doesn't much care how they are shaped.

While higher resolutions offer more information for the comparison, there is a performance penalty in processing all that information. Some OCR software limits the typesizes it can work with at various resolutions for exactly this reason. You don't need 90,000 bits of data to identify 72-point type.

Most OCR packages can handle varying formats easily—either automatically or by asking you to define manually which zones are to be read and in which order.

There are several areas that can cause OCR software problems. These include:

- Underlined text.
- Joined text.
- Broken text.
- Kerned text.
- Boldfaced characters.
- Italics.
- Photocopied text.
- Dirt.

11

Getting the Best Halftones

Today, a dollar or less will buy a priceless gift for your descendants, something once exclusive to royalty and the wealthy: a realistic image of your humble self captured for posterity in a photograph. Until about 150 years ago, quality portraits required the efforts of an artist and were quite beyond the reach of the common person. In the 18th Century, it was unlikely that you knew what your grandfather or great-grandmother looked like unless you had met him or her in life. Family Bibles and other records could detail the names and statistics of our ancestry, but they provided no visual clues about how much we did—or did not—look like those who preceded us.

The most important first use of photography was as a personal family record, which is borne out by large numbers of Daguerreotype portraits of corpses in their caskets. Indeed, because early photographs were all one-of-a-kind, widespread dissemination of these images was almost impossible.

Mass Reproduction of Photos

The mass reproduction of photographs has been problematic for as long as we have had photography. The Daguerreotype was the first while-you-wait portrait. The original plate exposed in the camera was developed, framed, and presented to the client all in one visit. If you wanted several copies of a particular subject, you exposed several originals. Those without sufficient foresight to do that sometimes rephotographed an original to produce a copy, losing some quality in the process.

The invention of the negative in the late 1800's made it easier to duplicate photographs. The reversed, negative image could be used to repeatedly expose sheets of photographic paper, producing a quantity of prints, each of the same quality. Photographs frequently found their way into books in the 19th Century (and later) as pasted-in plates. This method was obviously more costly than printing illustrations from engravings, but there was no other way to reproduce the continuous tones of a photograph in quantity. For run-of-the mill books and periodicals, including actual photographs was too costly. Moreover, the time required to include pictures by this means was impractical when very large quantities were needed on short notice, as for newspapers and magazines.

Ignoring the rotogravure process for the moment, printing presses, quite simply, can't print anything other than solid tones. Most frequently, black ink is used to print black text and line art. Each additional color that is desired must be produced by an additional run through the press (or by a continual passage through several printing operations consecutively). In all cases, though, the ink laid down on the paper or other substrate is of a solid tone.

The Halftone Process

The halftone process was developed as a way to simulate gray tones with solid-color reproduction systems. A continuous tone image is divided into fixed-size cells, which might look like a window screen if their boundaries were superimposed on the image. These cells, however, are largely imaginary. Each is either empty or filled with a dot of a fixed shape but variable size. Halftone dots often appear round but may actually be elliptical or square.

The size of the dot determines how the eye perceives it. If the dot fills up 25% of its cell, for example, the eye will blend that dark area with the surrounding white substrate and perceive a light gray tone. If the dot is larger, say filling up 75% of its cell, the eye will see a much darker gray, approaching black. The number of cells per inch in our window screen determines the screen ruling or frequency of the halftone.

A halftone is a continuous tone image that has been converted to an array of these dots. Because the dots are all solid, the printing press (and, as we'll see, other solid color systems like laser printers) can handle them with no trouble at all.

Well, that isn't precisely true. Some of the problems that printing presses have with halftones can be important to you as a scanner user, because there are things you can do electronically with your scanned image to compensate for them. Therefore, you need to learn about two kinds of halftones: the electronic kind that you produce with your image editing or page layout software and then output on a computer printer and the traditional kind which may also result from your desktop publishing efforts. We'll look at traditional halftones first. Understanding how they are produced can give you a better feel for the corrections you can make.

Photography has always been a mixture of art and science. Capturing an image on film demands a special combination of chemistry, optics, and mechanical engineering. Deciding what to capture is something of an art; the best equipment in the hands of a person who lacks training or an artistic eye will produce nothing more than sharp, well-exposed snapshots.

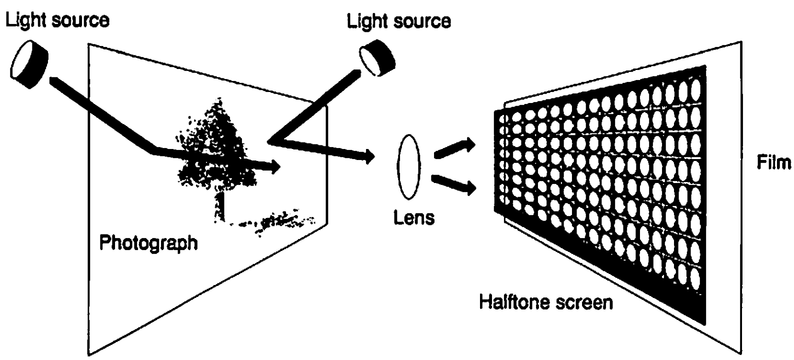
So, too, has halftoning been an art as well as a science. There are well-documented techniques for making halftones using conventional methods, yet, there has always been room for the craftsperson to apply experience and expertise to the process to produce a better halftone.

How Photographic Halftones Are Made

A photographic halftone is produced by placing the continuous tone artwork in the copyholder of a graphic arts camera (also called a process camera) and photographing it onto a piece of high contrast film that records only blacks and whites. The screen is placed between the original and the film, in close contact with the film. The halftone screen contains a pattern of out-of-focus dots that are reproduced on the film in varying sizes, depending on the amount of light reflected by various portions of the original continuous tone image. Figure 11.1 illustrates the process.

A dark area in the original will reflect little light, producing only a small dot on the halftone negative. Most of the area in a dark cell will, therefore, remain clear. Conversely, light areas in the original will produce larger dots in the halftone negative. The processed film will resemble a black-and-white negative (where the light and dark tones are reversed) with the screen's dot pattern superimposed.

Figure 11.1



This diagram shows how halftones are made photographically. The original photograph, at left, is placed in the copyboard of a process camera, and illuminated by a reflective light source. The light is focused through a lens onto a sandwich that consists of a halftone screen and a piece of film. The continuous tone image is broken up by the halftone screen, producing a representation of the original in the form of a pattern of dots. The more exposure received through a given dot, the larger that dot is.

Originally, halftone screens were made of two glass plates, each of which was scored, or ruled, with fine etched lines. The lines were then filled with a black or magenta material, and the two sheets of glass were placed at right angles to form a grid. The areas between the lines formed transparent squares, with the width between the lines referred to as the screen ruling, which varied between about 65 lines per inch and 150 lines per inch or more.

Glass screens are not mounted in contact with the film but are, rather, placed in a holder slightly in front of it. The screen distance affects the shape of the dots and the contrast of the final image, so glass screens are trickier to use than the contact screens which largely supplanted them.

As you might guess, contact screens are placed in direct contact with the light-sensitive film. Several different types are in common use. So-called square dot screens produce tiny squares that grow larger with exposure until their corners touch when they fill 50% of the cell, producing a checkerboard pattern. With more exposure, additional corner area overlaps, leaving small white dots until the entire cell is filled.

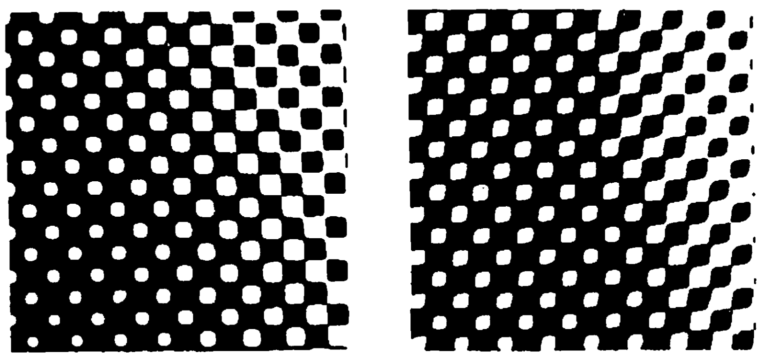
Elliptical dot screens use a slanted, oval shaped dot. These touch only at opposite corners when filled to 50%. Because the corner overlap is thus more gradual beyond this point, elliptical dot screens provide a smoother tonal reproduction in the midtones and are good for subjects with a lot of detail in those areas (flesh tones are a good example). Figure 11.2 shows examples of 50 percent dots in both eliptical and square dot configurations.

Contact screens are available in both positive and negative versions, so you can make halftones of both positive and negative originals.

The reason that making conventional halftones requires a bit of skill is that calculating the exposure is not as simple as it might appear. You want the halftone negative to include as much detail in the shadows (the clear areas with small, almost pinpoint black dots in them) as possible while keeping the highlights from filling up completely with filled black cells.

Photographic film requires that each exposed area receive enough light to pass the threshold needed to convert the silver halide crystals into metallic silver that can be developed and used to form an image.

Figure 11.2



A square dot screen (left) and eliptical dot screen (right). The dots that just touch the corners of each other (towards the upper right of each example) are 50 percent dots. Those at the lower left are larger than 50 percent.

Therefore, halftones frequently receive three separate exposures to insure that all areas of the image receive the right amount of light.

The first, or main, exposure is a photograph of the original copy through the lens of the camera with the contact screen in place. It provides the basic highlight, middletone, and shadow detail. This exposure is frequently insufficient to expose the pinpoint size dots in the shadow areas. So, a second, flash exposure is made by opening the camera back and exposing the film to a flash of light through the contact screen. This is a non-image exposure because the light is not reflected from the subject through the lens. In other words, all portions of the screen receive exactly the same exposure, regardless of how much light would have been reflected by the subject. The highlight areas, which are already fully exposed, are unaffected by this flash.

The shadow areas are given a bit of extra exposure that, when added to what they received from the main exposure, moves them past the threshold needed to produce a dot. Middletones are slightly affected by the flash. Calculating the right amount of flash exposure requires experience and expertise and makes producing halftones somewhat of a craft.

The final exposure is an optional "bump"exposure—a screenless exposure that boosts highlight contrast. The bump exposes all of the film to the same amount of light, supplying high contrast originals with important highlight detail. Bump exposures are very short—typically no more than 10% of the main exposure.

Electronic Halftones

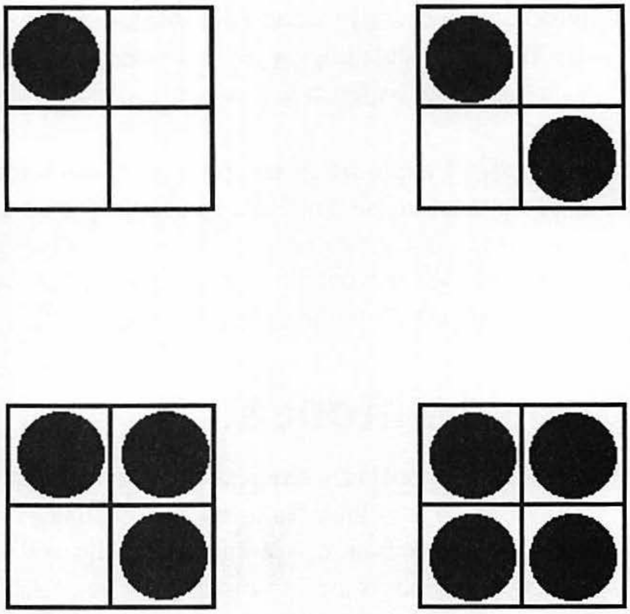
Electronic halftones duplicate the effect of photographic halftones and share some of their features and components, although an entirely different system is used to produce them. Instead of a camera, lens, and screen, a scanner is used to capture electronic halftones. Some of the early scanners used in the graphic arts did make use of a screen; the image was scanned through it to produce the halftone effect directly.

However, as software became more sophisticated, additional flexibility was provided by using the computer to calculate a wider range of screens by electronically dividing the image into the pattern of dots that we think of as a halftone. On one hand, this system provides a great deal of flexibility; to use a dot of any shape, you need only the proper software to make the necessary calculations.

On the downside, the resolution of the dot pattern is limited by the resolution of the output device used to write the final halftone. To produce a halftone, a 300 dpi printer must build larger halftone cells from its own tiny dots. The more of these dots required to build a halftone dot, the lower the resolution of the halftone image.

Figure 11.3 shows the smallest halftone cell possible—one that measures two dots on a side. Five different combinations of dots can be used to make a printer cell, with zero to four dots in each. These combinations yield 0%, 25%, 50%, 75%, and 100% halftone dots (and

Figure 11.3



This 2 x 2 printer cell allows four different shades of gray, plus white, and allows a halftone screen resolution of 150 lines per inch with a 300 dpit printer.

rather oddly-shaped ones at that). Thus, a 2 x 2 cell can represent four different shades of gray, plus white.

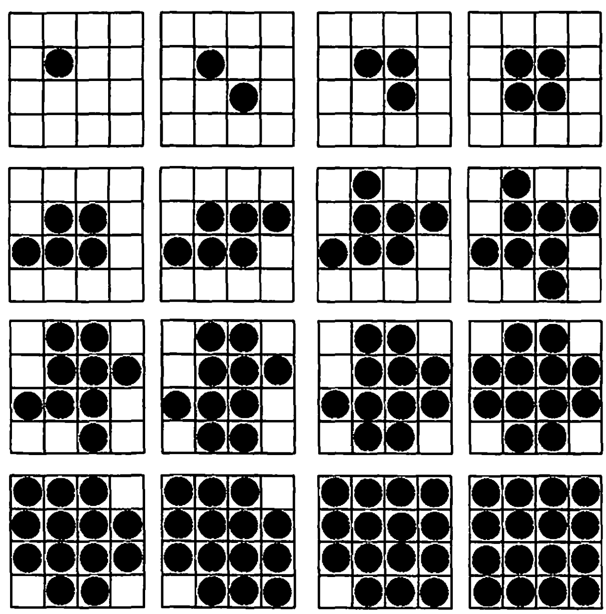
Note, too, that because two printer dots in each direction are required for each cell, the effective resolution of your printer is halved. A 300 dpi printer can represent only 150 2 x 2 printer cells per inch, producing a halftone screen resolution of 150 lines per inch.

In Figure 11.4, you see a larger, 4 x 4 printer cell, which can produce 16 different combinations, plus white. The gray scale has improved a little, but the halftone screen resolution has dropped to 75 lpi—about what newspapers use for their relatively coarse illustrations.

To achieve an acceptable number of gray tones, most desktop publishing packages default to a 6 x 6 halftone cell, which reproduces 36 gray tones, plus white, at a screen ruling of about 50 lines per inch. That may seem very coarse, but it works well for many applications.

The solution, of course, is to use a higher resolution output device. Even 600 x 600 resolution allows 64 gray tones at 75 lpi, while an imagesetter with 1270 dpi resolution lets you reproduce a full 256 tones

Figure 11.4



This 4 x 4 printer cell can reproduce 16 different size halftone dots, but resolution drops to 75 lines per inch with a 300 dpi printer. Even with this concession, 16 grays, plus white, are too few to reproduce most continuous tone images.

with a 75 lpi screen. If you plan to use a large number of detailed photographs in your publication, you will have to send your output to a service bureau with an imagesetter, make halftones the conventional photographic way, or settle for less than optimal results.

Getting the Best Halftones

Creating a good electronic halftone requires more than just choosing a suitable halftone screen pattern. Optimizing your halftones is a complex process that entails four different steps

- Optimizing the original image.
- Optimizing the electronic image.
- Optimizing the halftone pattern.
- Optimizing the printing process.

This section will look at each of these steps and provide some advice you can use to fine tune your own electronic halftoning.

Optimizing the Original Image

Obviously, you'll want to start off with the very best original image you can get. Any reproduction process necessarily costs you some quality at each step, and while you may be able to make some adjustments that make an electronic image look better than the original, the information needed to do those manipulations must be present in the original photograph. You may be able to improve contrast or sharpen details, but you inevitably lose something else in the process.

There will be times when you simply must work with a given image for some reason, usually because the camera negative is not available. In those instances, you will have to rely on the other three steps in the process.

Use a Large Size Print

If possible, you should start with a high quality 8 x 10-inch print. It doesn't matter whether the finish of the print is glossy or matte. You can use a smaller print—5 x 7 or 4 x 5, for example—and still get good results. But there are several reasons to use an 8 x 10 if possible.

- Larger prints have more detail. You may not need all that detail, and you will probably discard a great deal of it as the photograph is printed, but it's nice to have. Extra detail gives you more flexibility in choosing the area of a photograph you want to use. If you need to pull out a single face from a group, you can do it, even with a relatively low resolution 100 to 200 dpi scan. The exact same image in 4 x 5-inch size won't have sufficient detail for maximum enlargement. Even a 400 dpi scan can't pull out detail that doesn't exist in the original.

- Larger prints have a longer tonal range, particularly when made from larger negatives. Take two identical pictures, one made with a 120 format camera and one with a 35 mm camera, each equipped with a normal lens, and using the same film. Make an 8 x 10 print of each. The one made from the 120 format negative will be sharper, of course, but it will also offer more gray tones; the extra sharpness allows the film to record more gray tones in a given area of the subject. Gray scale is, after all, just another aspect of image resolution.

  If you make an 8 x 10 and a 4 x 5 from the same negative, the larger print will have a longer gray scale for the same reason. More detail produces a greater selection of grays in your image.

Some Exceptions

Now that I've said that, let me contradict myself a little. The photographic process is so sharp in the first place (thousands of dots per inch versus hundreds with electronic systems) that you should not be deterred from working with small prints if you have to. A 4 x 5-inch print may be plenty sharp if you're planning to reproduce the whole thing at the same size or smaller.

In fact, I have frequently made successful scans from contact sheets made from the original negatives when the negatives themselves or larger prints were unavailable. I've had the most luck with contacts of 120 format film, which measure 2.25 x 2.25 inches or 2.25 x 2.75 inches. On occasion, I have even been able to use 35 mm contact sheets (on which the images measure only 1 x 1.5 inches!) when the originals were very, very good.

Not a lot of care is usually put into making contact sheets, because they are used only to select frames for enlargement. Further, since the exposure and contrast of each frame in a roll may vary (especially if the photographer brackets exposures), the quality of individual images on a single sheet can vary. Your results may be usable, or not, so count on this method only as a last resort.

Watch the Contrast of the Original

The reproduction process tends to increase contrast. Therefore, you may have better results if you start with an original photograph that is slightly flatter than you might like. You can use your image editing software to beef up the contrast when you fine tune the image.

Gray maps and other image editing tools are not covered in detail in this book (there's a whole chapter on the topic in The Scanner Handbook), but as you work with images you'll get a feeling for what types of photographs lend themselves to successful tweaking after a scan.

If you have a choice, you can specify a slightly lower contrast print when your photographs are made. These are produced by exposing the print onto a lower contrast grade of paper (say, Grade 1 instead of Grade 2) or by using a lower graded filter when the prints are made on multicontrast paper. Talk to your photo lab to see what your options are.

Look Out for Dust, Scratches, and Other Artifacts.

You can actually touch these up more easily with your electronic image editing software than you can using traditional spotting methods, but it's nice to start out with a clean photo if you can. Your image software can pick up tones from one area and drop them on a damaged area of your photograph, but it can't replace any details that are obscured by artifacts.

Optimizing the Electronic Image

You can accomplish this by getting the best capture of your photograph possible. Part of that involves choosing a suitable scanning resolution and remembering that the highest possible resolution isn't always necessary or desirable

You also should watch out for careless techniques that can rob you of detail. For example, the platen of your scanner should be clean and the device located in a vibration-free environment. These are common sense rules that I won't repeat here.

The other thing you can do is to manipulate the captured image. You can adjust the gray map so the available gray tones are distributed to the parts of the image that has the details you want to retain. That is, it makes little sense to spread 64 tones equally across the scale if most of the important information in your photograph is in the shadows. Refer to *The Scanner Handbook* for more information on working with gray maps.

You might find it useful to scan a gray scale at the same time you scan your photograph, since the gray scale has gray tones of fixed, known values. You can then use the scale to calibrate your scanner with your video display and printer, so you'll know just what tones are produced under your particular configuration. Picture Publisher and several other packages furnish a gray scale you can use, and photography stores sell ones that will work well with most scanners.

Optimizing the Halftone Screen

You can do quite a bit to optimize your halftone by selecting the correct halftone pattern and screen angle. Not all screens at a given ruling will produce the same number of gray tones, sharpness, or visual effect. This next section will explain some of the differences you need to be aware of.

Types of Screens

The screen you select for your halftone can have an effect on the appearance of the final image. Some types of screens are better for particular types of images. Examples of some of the halftone screen variations you have to work with are shown in the section that follows The advantages and disadvantages of each type are discussed below.

Bayer screen. This is a popular halftone pattern that is neutral toward vertical and horizontal lines and is therefore a good choice for subjects that have both. While fine detail is preserved with this criss-cross pattern, contrast suffers, so you may want to make some adjustments with your image editing program before you start to scan. Keep in mind that this pattern looks better on a CRT screen than it prints, so you might want to make a few test prints before committing yourself to it in a publication. A Bayer screen is shown in Figure 11.5, which has been slightly enlarged so you can see the screen pattern.

Figure 11.5

Bayer halftone screen

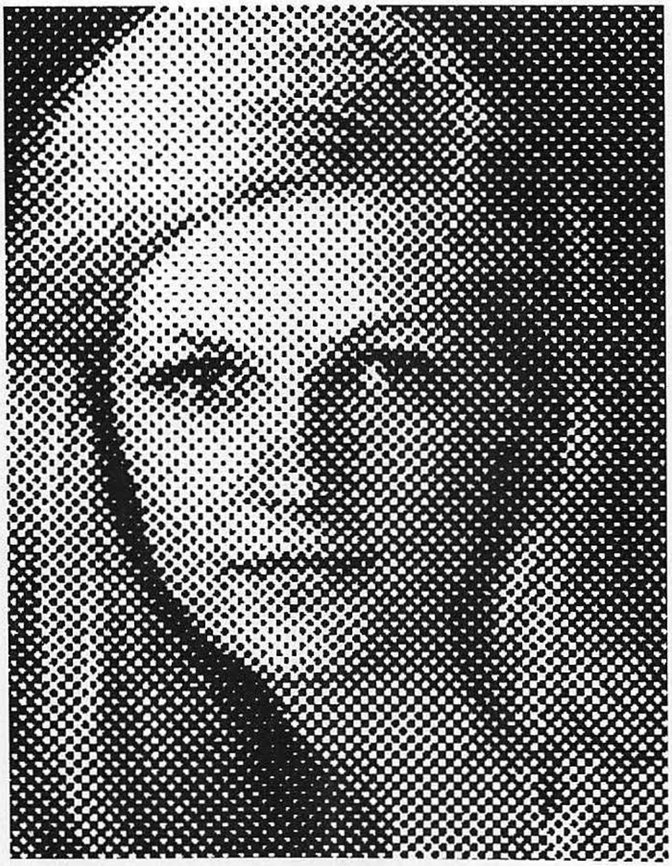
Fine fatting screen. This is a variation on the spiral pattern, described below, using smaller dots that provide a sharper image. It's a good choice for images with higher contrast and small details that are important. An example of a fine fatting screen is shown in Figure 11.6.

Spiral screen. This pattern, also known as Coarse Fatting in some software packages, is an excellent choice for laser printers, because it provides a longer gray scale at a slight sacrifice in resolution. A spiral screen is shown in Figure 11.7

Figure 11.6

The fine fatting halftone screen pattern is provided by most scanning and image control programs, and is excellent for images with fine details and high contrast.

Figure 11.7

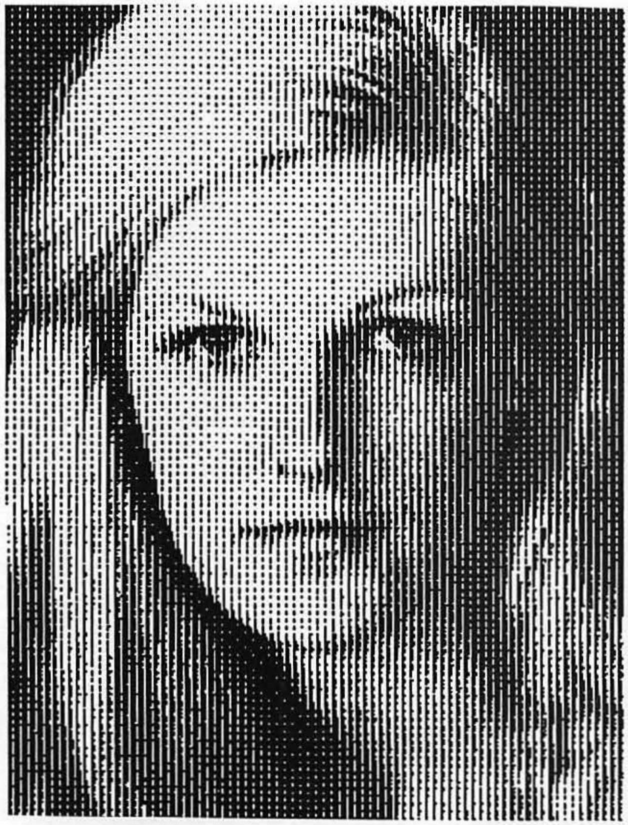


A coarse fatting or spiral screen pattern, which works well with the relatively low resolution of laser printers.

Horizontal line screen. This screen breaks the image into a series of horizontal lines with embedded dots that are used to convey the gray scale information. It is an excellent choice for images that have strong horizontal lines; the pattern helps preserve detail in that direction. Conversely, vertical lines are disrupted by a horizontal line screen, with a loss of resolution.

Vertical line screen. A vertical line screen is a horizontal line screen turned 90° It might be used for images with strong vertical lines, such

Figure 11.8



Vertical line screen

as buildings, groups of trees, and, oddly enough, faces. Horizontal lines aren't captured well with this screen. A vertical line screen is shown above in Figure 11.8.

Halftone Screen Angles

If the halftone dot pattern happens to align exactly with important lines in your image, the effect will be disturbing to the eye. Many of the things we photograph have vertical or horizontal lines, because up is the direction things tend to grow in nature, and humans tend to build things that are square and perpendicular. So, it's common to tilt a halftone screen at an angle to keep the dot pattern from interfering with the natural lines of the subject. A 45° angle is commonly used. In color printing, a variety of screen angles is used to allow the printing of several different colors without all the dots being superimposed on one another.

For many halftones, you'll want to stick with a 45° angle. Vertical and horizontal line screens, by definition, look good when oriented up-and-down relative to your image. However, these screens, too, can be set on an angle, producing an interesting effect. Experiment until you know what looks best with each type of image you scan.

Optimizing the Printing Process

As already noted, the resolution of the final output device can have a dramatic effect on the quality of your halftone. If possible, you should use a destination printer or imagesetter with the highest resolution justified by the application. That is, 300 dpi printers won't reproduce very many gray tones, but they can probably do a good enough job for a low budget, short run desktop publication.

You can also direct your output to a printer or imagesetter designed from the ground up to provide higher resolution, but that usually raises costs. You may not need, or be able to afford, 2570 dpi resolution, for example, but you might be able to get along just fine with 1200 dpi output at a fraction of the cost.

Compensating for Dot Gain

In my opinion, it will still be quite some time before desktop publishers are making the majority of the halftones used in really sophisticated desktop publications. As I've noted before, producing halftones photographically is still the cheapest way to do it and, given an experienced hand, the method that produces the best quality.

To see why, you need look no further than the topic of dot gain, which is the enlargement of printed halftone dots caused by spread under the pressure of impression, the absorbency of the paper stock, and any slur or doubling of dots that results from the reproduction process. In effect, a dot prints larger than it appears in the original halftone.

Dot gain isn't uniform for all dot sizes. The very largest dots don't have much room to grow, so you won't see the effects much in the shadow areas of an image. The very smallest dots don't lay down very much ink on the paper, so there is less to spread. So, it turns out that dot gain is greatest in the middletone areas and least in the highlight and shadow areas.

That means you can't compensate for dot gain by making all the dots in your halftone smaller to begin with. Instead, you must somehow relate the amount of gain to the image you halftone, adjusting only the middle tones of the picture to allow for the growth of the dots on the press.

If I had to tackle something like that, I'd want to have a densitometer (a device that measures and reports the relative amount of light transmitted through a piece of film) and a conventional halftone handy. If you're doing a great many halftones, have access to a high resolution imagesetter (so the work is worth the effort), and are willing to experiment, you can probably gain some experience in estimating how much you need to alter your midtones to compensate. I recommend printing up some test sheets with sample images that have had their gray maps manipulated in various ways. Then you can compare the printed sheet with the original to see what changes improved the image most. Keep in mind that variations in paper stock can cause variations in dot gain as well.

Halftones from Pre-Screened Images

The most informal of publications frequently are faced with the most challenging of scanning problems: how to capture and reuse an image that has already been screened and printed. More sophisticated publications occasionally face this problem, but usually, they are working from original photographs. It is the poor publisher of a limited-circulation in-house newsletter who wants to reuse a photo clipped from another publication or one that is for some reason not available as a continuous tone photograph.

Let's leave copyright considerations aside for the moment and just learn what to do. You have two main problems to overcome. First, the printed copy, since it is already printed, probably has gained a bit of contrast over the original photograph. Scanning, screening, and printing it again will increase contrast further.

Second, the printed copy already has a screen pattern of an unknown screen angle and ruling. These factors and the size of the printed piece may make the photograph unsuitable for reuse.

In addition, the printed image has suffered from dot gain, slurring, smeared ink, and other defects which your copy can only hope to duplicate. Fortunately, there are several methods you can use to salvage the image.

- If the printed copy is a high quality halftone with crisp, well-defined dots, you can scan it at a resolution that will allow you to reproduce the image in your publication at the same size. You'll be, in effect, reusing the halftone screen that was applied in the first place and not introducing a second screen on top of the first. This technique is known in the graphic arts industry as copy-dot, because you simply copy each of the dots in the original.

  Note that you can't enlarge or reduce the image significantly. If you make the image larger, the halftone dots will grow visible and probably be objectionable. Shrinking the image will cause many of the dots to merge and form undesirable patterns.

- You can try to guess the screen angle of the original (it is probably 45° for a monochrome image) and then scan and apply a new screen at about 30° from that original angle. If you are close enough, the results may be acceptable. If the two screens clash, though, you'll end up with a moire or rosette pattern that will probably ruin your image.

  In that case, try rescanning the image at a slightly different angle. Skewing the printed piece on the platen by only a few degrees can produce dramatically different results. If the picture is important enough to you, it may be worth the effort to experiment.

- If the printed piece is in color, you may have a much better chance of rescanning it. Because color photographs are printed with four overlapping screens, the dots tend to blur together when scanned with a monochrome or gray scale scanner. The scanner, like your eye, just may see the picture as a continuous tone image and scan it like one (with only a faint underlying texture of dots). You can then rescreen the image successfully, particularly if you use a coarser screen ruling.

- Try blurring the image to throw the original halftone pattern out of focus. You lose some image quality and resolution, but you avoid moire patterns. Most image editing software provides a blur option you can use to smear the entire image slightly. You can also experiment with placing an acetate sheet between the original and the platen, moving it far enough away from the sensor to blur the image.

Calculating What Screen Ruling to Use

Often, you'll want to select a particular halftone screen ruling to match the number of gray tones in your scanned image and the output resolution of your printer or imagesetter. That is, you know roughly what the tonal range of your image is—whether it is a high contrast image with a relatively small number of gray tones, a lower contrast image with many distinct tones, or something in between.

You also know what the resolution of your output device is. If you're using a page printer, there's seldom any reason to use less than the full 300 dpi resolution of the device. Your only choice may be between several different resolutions of imagesetter.

Given those fixed values, you'll want to calculate what halftone screen ruling to specify to your desktop publishing program, image editing software, or scanner (if you're halftoning during the scanning operation).

The formula used to calculate the screen value is a simple one:

Screen Resolution=DPI of output device/sq. root of gray levels

So, if you're using a printer with a resolution of 600 dpi and you want to reproduce 64 different gray levels, your ideal halftone screen will be 75 lines per inch:

600 /SQR(64)=75.

Calculating the Gray Tone Capabilities of a Given Screen Ruling

Sometimes you'll be locked into a particular screen ruling or range of rulings, either because your software doesn't allow you to specify other settings or because you are using a paper quality that doesn't lend itself to fine screen rulings. (Cheap, absorbent papers like newsprint, for example, don't allow fine screen rulings because the ink spreads, enlarging the size of the dots.)

The formula used to figure the number of gray levels is as follows:

Gray levels=(Printer Resolution in DPI/Halftone Screen Resolution)^2

For example, if you are using an imagesetter with a resolution of 1200 dots per inch and you want to use a 100 lpi screen, you'll be able to reproduce 144 levels of gray:

$(1200/100)^2$=144.

Figuring Scanner Resolution

As I've mentioned several times in this book, you'll usually want to avoid scanning at a resolution that is either too high or too low for your application. If the resolution is too low, you fail to capture some useful information. If the resolution is too high, your final image isn't improved appreciably, and you lose valuable time and disk storage space.

There's not much reason to scan a photograph at more than 125% of the final printer resolution. You may see some small improvement if you go up to 250% of printer resolution, but anything beyond that is wasted.

For example, assume that you have a photograph that is 4 inches high and you want to reduce it by 50% to 2 inches in your publication. You're sending the files to an imagesetter that allows a reasonable number of gray values at a 150-line screen resolution. Therefore, you need scan at no more than 100 lines per inch to get the quality you want.

Suppose you wanted to scale the image up 50% instead of down, so it would appear 6 inches high in the publication. At the same 150-line halftone screen, you would need to scan at 300 dpi.

You may find that this formula seems to be telling you that you need a lower halftone screen resolution, particularly when scaling images up at higher rulings. Your scanner may not have a resolution high enough to produce the recommended resolution. In such cases, your scanner's ability to interpolate higher resolutions (up to 400 or 600 dpi with some models) may help. Or, you may indeed be better off with a lower resolution screen, because the higher one may not provide a sharpness advantage and always costs you some gray tones.

The actual formula used to calculate scanning resolution is a simple one: the resolution you need is the maximum screen ruling you plan to use multiplied by the square root of 2 (1.414).

Summary

Considering the importance of visual information in our society today, the mass reproduction of photographic images is a precious gift—one that has been extended to those of us with desktop publishing and scanning equipment. Today you can duplicate images that couldn't have been purchased for a king or queen's ransom 300 years ago.

While the invention of the negative in the late 1800's made it easier to duplicate photographs in larger quantities, true mass reproduction was impossible until the halftone process was developed as a way to simulate gray tones with solid-color reproduction systems.

In halftone reproduction, a continuous tone image is divided into fixed-size cells. Since the dots are generally very small, the size of the dot determines how the eye will perceive it. If the dot fills up 25% of its cell, the eye will blend that dark area with the surrounding white substrate and perceive a light gray tone. If the dot is larger, say filling up 75% of its cell, the eye will see the area as a much darker gray.

The number of cells per inch determines the screen ruling or frequency of the halftone. A halftone is a continuous tone image that has been converted to an array of these dots.

A photographic halftone is produced by placing the continuous tone artwork in the copyholder of a graphic arts camera and photographing it onto a piece of high contrast film placed between the original and the film. The halftone screen contains a pattern of dots that is reproduced on the film in varying sizes, depending on the amount of light reflected by the original.

There are several different types of contact screens in use, including square dot screens, which produce tiny squares that grow larger with exposure until their corners touch when they fill 50% of the cell, and elliptical dot screens, which use a slanted, oval shaped dot. Several exposures are often required to produce a conventional halftone.

Electronic halftones duplicate the effect of photographic halftones, using an entirely different reproduction system. Instead of a camera, lens, and screen, a scanner captures electronic halftones.

To produce a halftone, a 300 dpi printer must build up larger halftone cells from its own tiny dots. The more of these dots required to build a halftone dot, the lower the resolution of the halftone image.

Creating a good electronic halftone requires more than just choosing a suitable halftone screen pattern. Optimizing your halftones requires optimizing the original image, the electronic image, the halftone pattern, and the printing process.

For example, you can use larger prints, manipulate the scanned image to optimize placement of gray tones, and select the halftone screen that will best reproduce the image. Then, the output can be directed to a high resolution output device.

12

Color For Scanner Users

This chapter provides a general introduction to the color theories you'll need to be familiar with to use a color scanner successfully. I'll describe the most frequently used color models and explain the differences between the way scanned color is displayed on your CRT monitor and the way it is output by your printer. There will even be some tips for selecting the best monitor for your scanning work.

But first, let's look at the general topic of color and how it has come to be so important to computer users in general and scanner owners in particular. For those of us who waited until we got our first color TVs in the late 1960's to discover that Huckleberry Hound was blue, all the ruckus borders on the amazing.

Introduction to Color

Human perception of color is a strange and wonderful thing. Most of us remember a little of how the eye sees from our high school science class: the retina of the eye contains tiny rods and cones that somehow react to light and provide information to the brain which we interpret as sight.

Color vision derives from the three different types of cone cells, which respond to different wavelengths of light. Our eyes are able to detect only a relatively narrow band of frequencies, ranging from 400 nanometers at the short (violet) end of the visible spectrum to 700 nanometers at the long (red) end.

Artificial color systems, which include computer scanners, monitors, printers, and other peripherals, attempt to reproduce, or model, the colors that we see, using various sets of components of color. If the model is a good one, all the colors we are capable of detecting are defined by the parameters of the model. The colors within the definition of each model are termed its color space. Because nearly all color spaces use three different parameters, we can plot them as x, y, and z coordinates to produce a three-dimensional shape that represents the color gamut of the model.

The international standard for specifying color was defined in 1931 by the Commission Internationale L'Eclairage (CIE); it is a scientific color model that can be used to define all the colors that humans can see. However, color scanner users most often work with one of three or four other color models, which are more practical because they are based on the actual systems used to reproduce the colors. None of these systems can generate all the colors in the full range of human perception, but they are the models with which we must work. There are some efforts underway to define new models, but so far nothing that will completely take over the industry in the near future has appeared on the horizon. Your best bet as a scanner user is to learn about all of them, since most image editing and scanning programs will support any of the most-used color models.

Of the three most common models, the ones based on the hue-lightness-saturation (HLS) and hue-saturation-value (HSV) of colors are the most natural for us to visualize, because they deal with a continuous range of colors that may vary in brightness or richness. Unfortunately, we must deal with two other models, called additive color and subtractive color, which are easier for computers to handle, because the individual components are nothing more than three basic colors of light. Additive color is commonly used in computer display monitors, while subtractive color is used for output devices such as printers. Since you need to understand how color works with these peripherals, I'll explain the additive and subtractive models first.

Additive Color

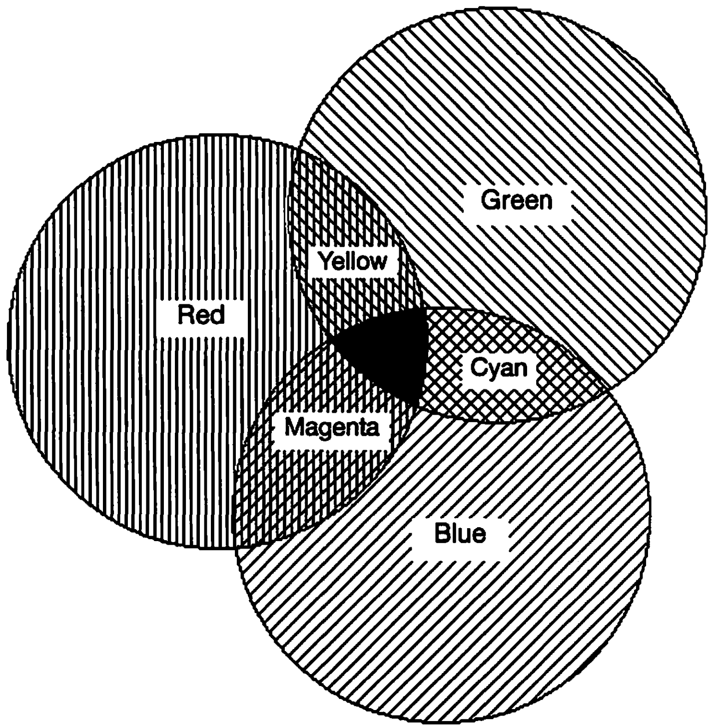
Color monitors produce color by aiming three electronic guns at sets of red, green, and blue phosphors coated on the screen of your CRT. The guns excite the phosphors in proportion to the amount of red, green, or blue light in a given pixel of the image. The phosphors glow, and our eyes add their illumination together, perceiving a color image. If none of the phosphors glows, we see a black pixel. If all three glow in equal proportions, we see a neutral color—gray or white, depending on the intensity.

Such a color system uses the additive color model—so called because the colors are added together. Additive color is explained in Figures 12.1 and 12.2 on the following pages. A huge selection of colors can be produced by varying the combinations of light. In addition to pure red, green, and blue, we can also produce cyan (green and blue together), magenta (red and blue), yellow (red and green), and all the colors in between. As with gray scale data, the number of bits used to store color information determines the number of different tones that can be reproduced

No CRT device available today produces pure red, green, or blue light. Only lasers generate absolutely pure colors, and they aren't practical for display devices. We see images through the glow of phosphors, and the ability of phosphors to generate absolutely pure colors is limited. Color representations on a monitor differ from brand to brand and even from one monitor to another within the same brand.

Moreover, the characteristics of a given monitor change as the monitor ages and the phosphors wear out. Some phosphors, particularly blue ones, change in intensity at a different rate than others. So, identical signals rarely produce identical images on CRTs, regardless of how closely the monitors are matched in type, age, and other factors. We'll look at this topic a little more later on when I explain a little about calibration.

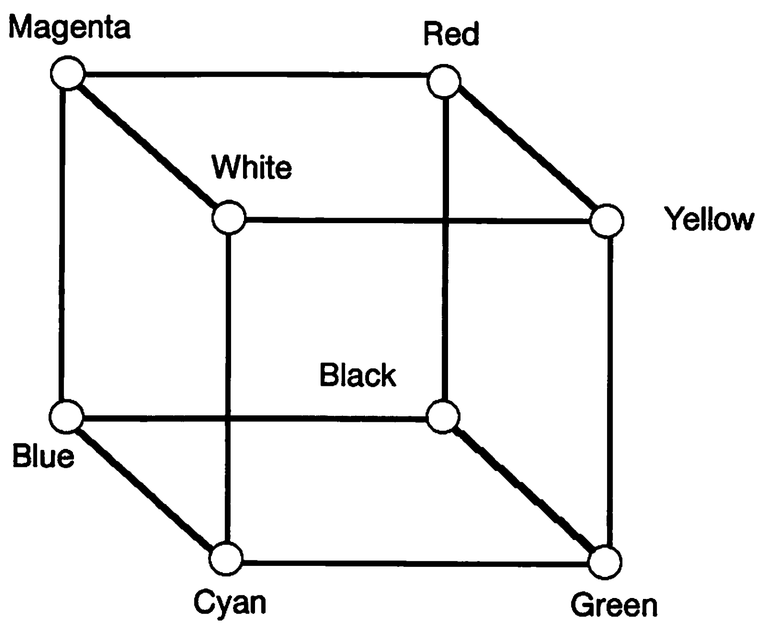
Figure 12.1



This illustration shows one way of thinking of additive color, in a two-dimensional color space. The largest circles represent beams of light in red, green, and blue. Where the beams overlap, they produce other colors. For example, red and green combine to produce yellow. Red and blue add up to magenta, and green and blue produce cyan. The center portion, in which all three colors overlap, is black.

However, this two-dimensional model doesn't account for the lightness or darkness of a color—the amount of white or black. That added dimension is dealt with in the model shown on the next page.

Figure 12.2



In a three dimensional additive color model, the color space can be represented by a cube like the one shown above. Red, green, and blue colors are positioned at opposite corners of the cube, with their complementary colors arranged between them. White and black are located opposite one another as well. Any shade that can be produced by adding red, green, and blue together can be represented by a position within the cube.

Additive Color and Your Monitor

In practice, most monitors display far fewer colors than the total of which they are theoretically capable. The number of colors that can be displayed is largely a function of the amount of memory available to store and manipulate color. Remember that your Mac must manage both the memory used to represent the current image and the temporary buffers used to store information not currently displayed.

A standard 640 x 480-pixel Mac II display may store 4 bits of information per pixel, for a total of 154K for the image and 154K for the buffer. Upping the ante to 8 bits per pixel increases the memory requirement to 614K bytes, while a full color, 24-bit image at even higher resolutions demands 5 or 6Mb.

New display cards and monitors capable of handling 24-bit color are becoming available. At first, they were rather costly—in the $3000 to $4000 range for the graphics card alone. That's because a lot of memory and special graphics coprocessors are needed to manipulate that much information at a speed we can use. Costs have dropped as technology advances.

The average Mac color user will probably use a dual-mode 8/24-bit color card at first. These will be lower cost (in the $600 range) graphics adapters like my RasterOps Colorboard 264 that can be easily switched from 8-bit mode (for general work, layout, and gross image editing tasks) to 24-bit mode for final evaluation and editing. Redrawing the screen is much slower in 24-bit mode—it may take 5 or 10 seconds to repaint an image—but we pay the price in order to gain true photorealistic color.

Subtractive Color

There is a second way of producing color that is familiar to computer users, and it, too, has a color model that represents the color gamut. When we represent colors in hardcopy form, the light source comes not from the image itself, as it does with a CRT image. Instead, hard copies are viewed by light that strikes the paper or other substrate, is filtered by the image on the paper, and then is reflected back to our eyes.

This light starts out with equal quantities of red, green, and blue light and looks white to our eyes. The pigments the light passes through before bouncing off the substrate absorb part of this light, subtracting it from the spectrum. The light that remains reaches our eyes and is interpreted as color. Because various components of light are subtracted from white to produce color, this color model is known as the subtractive system.

The three primary subtractive colors are cyan, magenta, and yellow, and the model is usually known as the CMY model. When black is added (for reasons explained shortly), it becomes the CMYK model (black is represented by its terminal character, k, rather than b to avoid confusion with the additive primary blue).

In subtractive output devices, cyan, magenta, yellow, and sometimes black pigments are used to represent the full gamut of colors. It's obvious why additive colors won't work for hard copies: it is possible to produce red, green, and blue pigments, of course, and we could print red, green, and blue colors that way (that's exactly what is done for spot color). However, there would be no way to produce any of the other colors with the additive primaries. Red pigment reflects only red light; green pigment reflects only green. When they overlap, the red pigment absorbs the green, and the green absorbs the red, so no light is reflected and we see black.

Cyan pigment, on the other hand, absorbs only red light. It reflects both blue and green, producing the blue-green shade we see as cyan. Yellow pigment absorbs only blue light, reflecting red and green, while magenta pigment absorbs only green, reflecting red and blue. When we overlap two of the subtractive primaries, some of at least one color still reflects. Magenta (red-blue) and yellow (red-green) together produce red, because the magenta pigment absorbs green and the yellow pigment absorbs blue. Their common color, red, is the only one remaining.

Of course each of the subtractive primaries can be present in various intensities or percentages, from 0 to 100%. The remainder is represented by white, which reflects all colors in equal amounts. The subtractive model is illustrated in the figures on the next two pages.

So, in our example above, if the magenta pigment was only 50% present and the yellow represented at 100%, only half of the green would be absorbed, while 100% of the blue would be soaked up. Our red would appear to be an intermediate color, orange. By varying the percentages of the subtractive primaries, we can produce a full range of colors.

Well, theoretically we could. You'll recall that RGB monitors aren't perfect because the color of the phosphors can vary. So, too, is it impossible to design pigments that reflect absolutely pure colors. Equal amounts of cyan, magenta, and yellow pigment should produce black. More often, what you'll get is a muddy brown. With many output systems, that's what you'll have to settle for. It's a complicated enough procedure to lay down sets of cyan, magenta, and yellow pigment in perfect register. Indeed, many daily newspapers that print color use this three-color system to this day.

However, better results can be obtained by adding black as a fourth color. Black can fill in areas that are supposed to be black and add detail to other areas of an image. While the fourth color does complicate the process a bit, the actual cost in applications like offset printing is minimal. Black ink is used to print text anyway, so there is no additional press run for black. Moreover, black ink is cheaper than critical process color inks, so it's possible to save money by using black instead of laying on three subtractive primaries extra thick.

The output systems you use to print hard copies of color images use the subtractive color system in one way or another. Most of them are unable to print varying percentages of each of the primary colors. Offset presses, inkjet printers, color laser printers, and thermal wax transfer printers are examples of these. All these systems must simulate other colors by dithering, which is similar to the halftoning system discussed earlier. A few printers can vary the amount of pigment laid down over a broader range. Thermal dye sublimation printers are an example of this type. These printers can print a full range of tones, up to the 16.7 million colors possible with 24-bit systems.

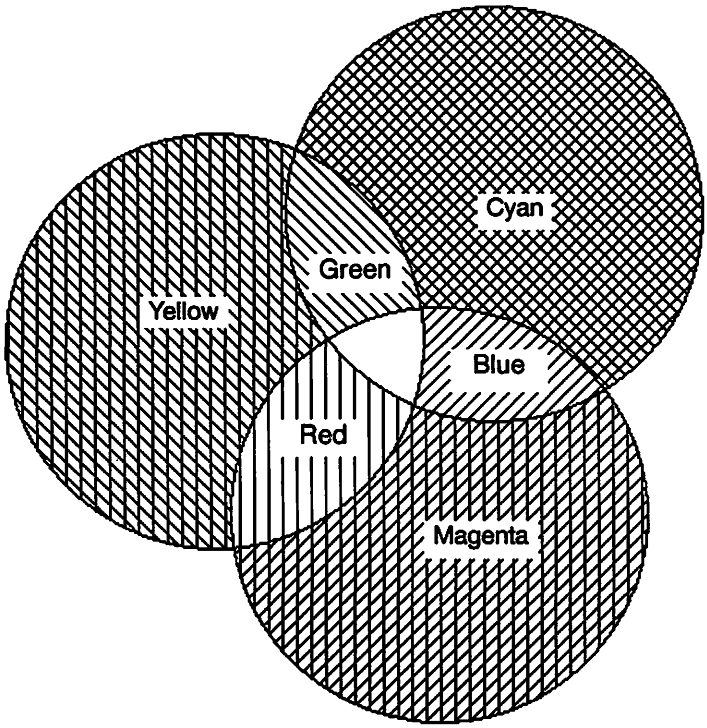
Other Color Models

Other color models that have been developed, include hue-saturation-brightness, known as HSB or HLS (for hue-lightness-saturation). You'll find HSB or HLS supported by many software packages.

In this model, individual colors, called hues, are represented as they are in a rainbow, as a continuous spectrum, arranged in a hexagon like that shown in Figure 12.3. The full color space is represented as a double hexcone that extends upwards and downwards from the hexagon, as shown in Figure 12.4. The top of the axis drawn through the center of the cones represents pure white, while the bottom point represents black. Moving higher in the cone produces lighter colors, lower in the bottom cone, darker colors. Saturation is represented by movement in a third direction, outward from the center axis. The center represents a desaturated color, the outer edges fully saturated hues.

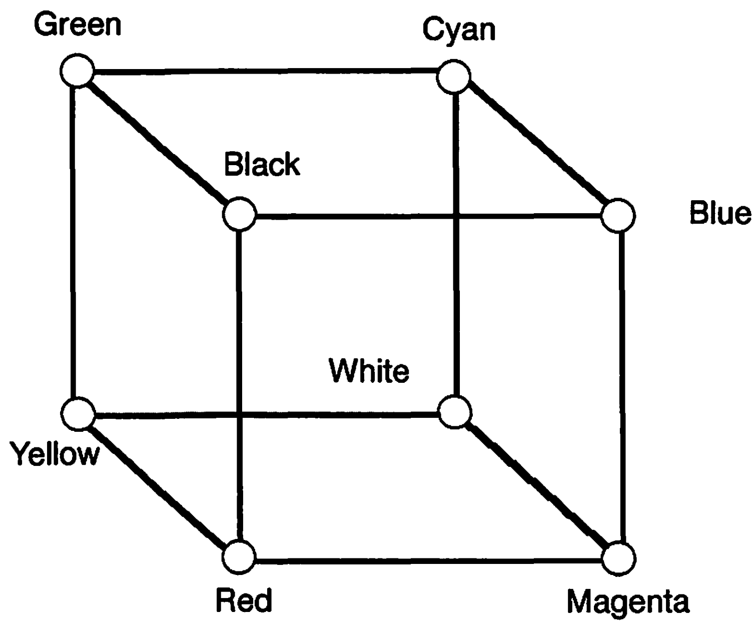
All colors can be represented by three parameters in this model. The hue is the particular position of a color in the color wheel. Hues are often represented by angles, beginning with red at 0° and progressing around in a counterclockwise direction to magenta at 60°, blue at 120°, and so forth. The saturation of the color is the degree to which that color is diluted with white. A hue with a great deal of white is muted; a red becomes a pink, for example. A hue with very little or no white is vivid and saturated. Brightness can be thought of as the degree to which a color is diluted with black. Add black to our desaturated red (pink), and you'll get a very dark pink. Various combinations of hue, saturation, and brightness can be used to produce just about any color we can see. The illustrations on the two pages that follow will help you visualize how this color model works a little more clearly.

Figure 12.3



The subtractive color model can also be represented in a two-dimensional way, as you'll see in the illustration above. The large overlapping circles represent pigments, which each reflect cyan, magenta, and yellow light, and aborb the other two. Where the three overlap in the center, all three colors are reflected, producing white. When cyan and yellow overlap, both colors are reflected (and magenta is absorbed), so we see the color green. Cyan and magenta absorb yellow, and reflect a combination we see as as blue. Similarly, magenta and yellow together absorb cyan, and produce red.

Figure 12.4



A three-dimensional cube can also be used to represent the subtractive color model. It looks a lot like the cube shown in Figure 12.2, only it has been rotated so that yellow, cyan, and magenta now occupy the X, Y, and Z axes that represent red, green, and blue in the additive model. Any color can be indicated by a coordinate in this color space.

The Pantone Matching System

The Pantone Matching System is Pantone, Inc.'s check-standard trademark for color reproduction and color reproduction materials. At least, that's the legal terminology that Pantone is quite zealous about insisting on from helpless journalists like myself. In simpler language, PMS is just another way of referring to colors.

Instead of using a set of descriptive parameters to define every possible color, PMS takes a different approach. Pantone has identified 747 colors and defined the formulas needed to produce pigments of those colors. Each color is given a number. You use a printed matching guide — actually an ink swatch book, which is available at most graphic arts supply stores — to select the color. The samples are printed on both coated and uncoated paper. You can select a color and specify it by number. Your printer can then use the PMS formula to mix the proper ink. This is extremely useful when you want a specific spot color for, say, corporate trade dress or product packaging.

As useful as Pantone colors are for spot color, they can't be used to print full color images; to reproduce 256 different colors, you would need 256 different inks and press runs. If you don't need an exact shade, it may be less expensive to let your printer use the three process colors to produce the spot color you want. That's because, using a specially mixed color requires cleaning the press and other additional make-ready steps.

You can also use the Pantone Matching System just as a way to specify colors. You can choose a Pantone color and then tell your printer to approximate it as closely as possible using the four-color process.

If the RGB, CYMK, and HSB color models produce continuous color models, the Pantone model might be thought of as discrete points of particular color within the larger color space that humans can perceive. While the Pantone color selection is huge, it can't represent all possible colors, as the other models can.

Screen Size

I was rather surprised when I brought my new Sony color television home earlier this year. The box was labeled "27 Inch Screen: 28 Inches in Canada." Now, I knew that Canada had switched to the metric system some time back, but it took me a while to determine how 27 inches (US) could equal 28 inches north of the border. Could Canada have devalued the inch?

The answer is that in the United States, televisions are measured according to the diagonal measurement of the actual screen visible inside its border mask. In Canada, it's the full size of the picture tube that counts.

The Canadian system seems to apply to monitors in both countries. A 14-inch monitor may have a screen with a diagonal measurement as small as 12 inches to as large as 13 inches. You'll want to compare actual picture size when determining which screen size you want.

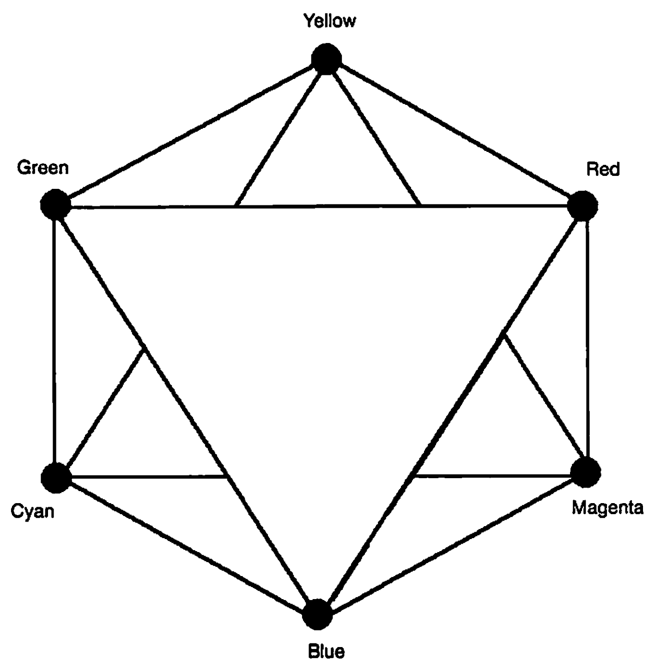
Make no mistake, at higher resolutions, screen size can be critical. When I first got a high resolution color card, I connected it to a Sony monitor with a 12-inch diagonal screen. Bad move. Most characters appeared so small that they were barely legible, despite the Sony's almost legendary sharpness.

I next decided to upgrade to a 20-inch color monitor to get the image size I needed. After a lot of research, I selected a top-selling model and proceeded to place a mail order. The dealer and I had reached the point of discussing shipping charges when he said something that brought me up short: it was going to cost $70 to ship the thing to me. When I asked him why, he started describing the monitor's weight and size: it was even larger than the 20-inch color television I have mounted on a big stand here in my office. There was no way I could fit that thing on my desk in front of my keyboard.

I should add that 20-inch color monitors are much bigger than my very compact 20-inch gray scale monitor; that was the scale I was mentally comparing my proposed purchase to. I had seen large color monitors at work in other folks' offices; I had just forgotten how huge they are.

It was at this point that I discovered the virtues of 16-inch monitors, like the Nanao Flexscan 9070U I eventually purchased. Its screen is large enough to display scanned images at a good size and produce legible text at high resolutions, and still it is small enough to fit on my desk.

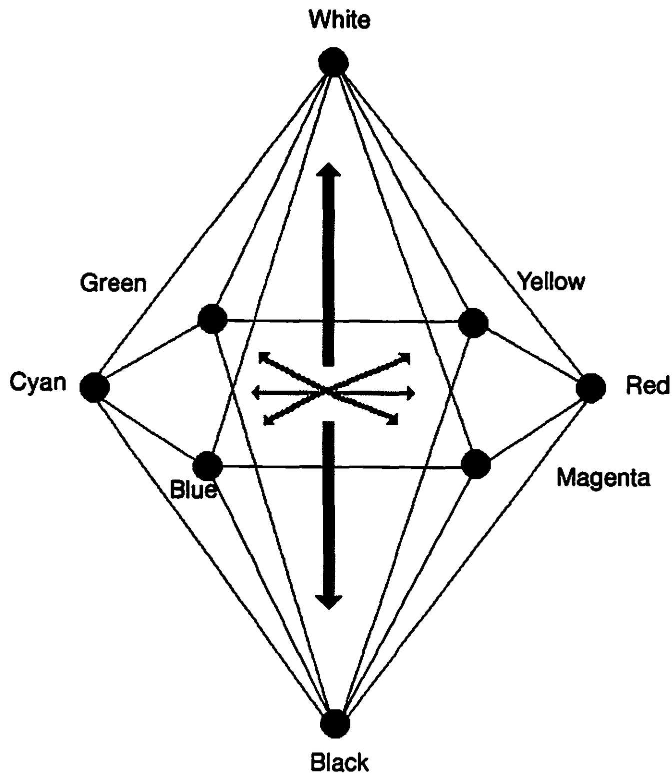
Figure 12.5



You can probably see traces of both the additive and subtractive color systems in this two-dimensional representation of the hue-saturation-lightness/brightness (HSL/HSB) color model. The downward pointing triangle within the hexagon includes the three additive colors, red, green, and blue. The upward-pointing triangle has the subtractive colors at each of its points.

In this model, each hue can be represented by a position along the perimeter of the hexagonal color wheel. If you want to account for the saturation and lightness or brightness of a particular hue, it's necessary to add a third dimension, shown in Figure 12.6.

Figure 12.6



When the hexagon on the previous page is turned on its side, in a manner of speaking, it forms the base for two hexcones that can represent all the colors in the HSB/HSL color model in three dimensions.

As before, individual hues can be found along the edge of the hexagon. Moving from that edge toward the center within the same plane indicates a color that is less saturated. The center of the hexagon represents zero saturation. The axis extending through the center of the hexcones represents lightness and darkness, with white at the top and black at the botton of the line.

Don't buy a monitor that is too small for the resolution you want to use, nor too large to be used comfortably.

Dot Pitch

The finer the dot pitch — the size of pixels on the screen — the finer the image. Monitors using Sony Trinitron technology provide the finest dot pitch — .25 millimeters. Other leading monitors offer .28 dot pitch. At the low end, a .31 pitch probably provides the coarsest acceptable dot size. There are even coarser pitches available in the extremely cheap monitors. Avoid them.

Other Factors

The other factors to consider include refresh rate (a 56 Hz rate can produce flickering; 72 Hz is unlikely to), the persistence of phosphors, and the amount of radiation emitted from the video display. It's unlikely you'll be able to tell much about these from the specifications published about a particular monitor. You're better off studying the reviews in InfoWorld and MacWorld to see how each stacks up against the others.

Not surprisingly, Windows 3.0 may be the driving force behind a general upgrading of some of these factors. Windows' white background color tends to accentuate flicker, since it involves all three electron beams in equal amounts. Moreover, a greater area of the screen is affected in Windows' graphics-intensive mode. Character-based applications typically illuminate a much smaller portion of the screen and are often refreshed at higher rates, so flicker isn't as noticeable.

In Europe, a 70 Hz refresh rate is standard, primarily because the higher rate reduces magnetic emissions. As more products are built to meet stiffer European standards, monitor users in other parts of the world benefit, too.

True Color and the Mac

My first Macintosh could display any color I wanted, as long as it was black or white. But true color has long since come to the Macintosh world. PC owners are ashamed it's taking them so long to catch up.

When the Apple Macintosh was introduced, it was capable of displaying only binary, black-and-white images. Macs had no color capabilities at all until March, 1987, when the Macintosh II was introduced. A month later, IBM debuted the Video Graphics Array (VGA) standard, which equalled or bettered the capabilities of the first Mac II. By all indications, the race for true color should have been a horserace from that moment on.

Yet, Macs have run away with the contest in the past few years, although the price tag for true color Macintoshes has bordered on the astronomical. Compared to $20,000 and up for a comparable workstation, a $12,000 full color Mac was a bargain but out of the price range of most of us.

Today, on a dollar for dollar basis, it is the PC that falls short. My Epson color scanner is connected to both the latest PC and a three-year old Mac II. The PC is equipped with a $2000 multiscanning monitor and one of the most advanced VGA boards available that doesn't include a graphics coprocessor. The Mac has a $1200 monitor/graphics card combination.

Yet, when I scan images into my PC, I am rewarded with a relatively coarse, dithered image that uses a mere 256 colors. The image displayed on the Mac looks almost exactly like the original photograph. The Apple system can display more than 16 million different colors. If this were a horserace, my PC mount would have stumbled at the starting gate.

Of course, a full color capable Mac cost a minimum of $5000 before January, 1991, when the Macintosh LC went on sale. A PC that was arguably the equal of the Mac in everything but color could be purchased for quite a bit less than $2000 at that time.

Summary

This chapter explains some of the concepts of color as they relate to the Macintosh and tells you some of what you need to know to do color scanning. Later chapters explore color separating and working with specific color-capable software packages.

Human perception of color is a strange and wonderful thing. Color vision derives from the eye having three different types of cone cells, which respond to different wavelengths of light. Artificial color systems, which include computer scanners, monitors, and printers, model the colors we see, using various sets of components of color in a color space.

Of the three most common models, the ones based on the hue-lightness-saturation (HLS) and the hue-saturation-value (HSV) of colors are the most natural for us to visualize. But, we must deal with two other models, called additive color and subtractive color, which are easier for computers to handle. Additive color is commonly used in computer display monitors, while subtractive color is used for output devices such as printers.

Color monitors produce color by aiming three electronic guns at sets of red, green, and blue phosphors coated on the screen of your CRT. The guns excite phosphors; if none of the phosphors glows, we see a black pixel. If all three glow in equal proportions, we see a neutral color—gray or white, depending on the intensity.

However, no CRT device today produces pure red, green, or blue light. Color representations differ from monitor to monitor, even within the same brand, and can change as the monitor ages. So, it's often necessary to calibrate your scanner, monitor, and printer so that the colors you see are close to what you get.

Subtractive color consists of the primaries cyan, magenta, and yellow, and the model is usually known as the CMY model. When black is added, it becomes the CMYK model.

In subtractive output devices, cyan, magenta, yellow, and sometimes black pigments are used to represent the full gamut of colors. The output systems you use to print hard copies of color images use the subtractive color system in one way or another.

Other color models that have been developed include hue-saturation-brightness, known as HSB or HLS (for hue-lightness-saturation). In this model, individual colors, called hues, are represented as a continuous spectrum, arranged in a hexagon.

All colors can be represented by three parameters in this model. The hue is the position of a color in the color wheel. The saturation of the color is the degree to which that color is diluted with white. Brightness can be thought of as the degree to which a color is diluted with black.

The Pantone Matching System (Pantone, Inc.'s check-standard trademark for color reproduction and color reproduction materials) is another common color model. Instead of using a set of descriptive parameters to define every possible color, PMS specifies a series of colors by number, and cross references printed ink swatch samples with the formulas necessary to reproduce those colors. As useful as Pantone colors are for spot color, they can't be used to print full color images.

The chapter concluded with discussions of the latest trends in software, displays, and display cards for producing 256-color and 24-bit color representations of images on CRTs.

13

How Scanners Capture Color

In the last chapter, I described in general terms how 24-bit true color has finally come to the Macintosh world. As you saw from the discussion of additive color and monitors, and subtractive color in output devices, there are many pieces that have to fall into place at about the same time for color to be really useful.

You must have a way to display color on the screen. You need software that can manipulate the colors in useful ways. Unless you want to call everyone over to view your CRT each time you refer to a color image you have created on your computer system, you'll also need a hardcopy output device. These technologies are becoming available.

Given that few of us are artists capable of creating all the original color images we need (or colorizing black-and-white images well enough to make them look professionally done), color scanners make up the final piece of the puzzle. Fortunately, a color scanner is a relatively simple piece of equipment, compared with a color monitor or a color printer, so scanner technology has been comfortably ahead of the other two fields for a long time.

Affordable Color Scanners

Color scanners have been available for more than 40 years. The first ones I saw cost more than a million dollars when you included the computer equipment needed to drive them. Today, color scanners are not only affordable, but the difference in price between them and gray scale scanners is evaporating. You may pay $1200 for a good gray scale flatbed scanner and $1700 for a similar model that can capture full color.

One of the top gray scale hand scanners costs $300; for another $200 you can have a color hand scanner. I can see the day when a scanner will be a standard peripheral included in the system price just as a mouse often is today.

The sharp reduction in the cost of color scanners makes it apparent that some other peripherals aren't matching those price/performance improvements. In fact, if we weren't still amazed by the color capabilities computers are gaining, we'd probably be outraged by the disparity in the prices of various pieces of equipment. For example:

- A color-capable computer system like the Macintosh LC costs $2000 or less.
- An entry-level color scanner can be purchased for around $500.
- A monitor capable of displaying the highest resolution color images your computer and scanner can handle costs between $3000 and $5000, including graphics card.
- The cheapest printers that can reproduce the full color gamut cost $5000 to $10,000 and up.

Isn't there something wrong here? Why should a whole computer with complicated mechanical systems like hard and floppy disks, expensive memory chips, and so forth cost a fraction of what is being charged for a glorified color television set? And why do printers cost so much, anyway?

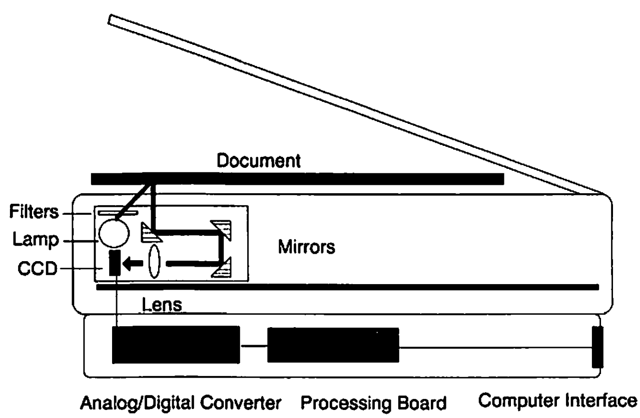
As a result of the continuing technology gap, you're likely to have a color scanner that is a lot more sophisticated than your display and output peripherals. In my case, I end up viewing dithered images on my monitor screen. I already have a few thousand dollars invested in CRT equipment without getting close to the 24-bit color price range, and I have no color output capabilities at all.

How Color Scanners Work

Now that you have a basic understanding of how colors are represented, you can readily see just how color scanners work: all you need is a system for capturing the proper amount of each of the primary colors of light in a given image.

Color scanners use three different light sources—one each of red, green, and blue—to scan a color image. To do so, some, such as the Microtek MSF-300Z, make three passes over the image—once for each color. My Epson color scanner does it differently, using three separate fluorescent tubes. The three alternate illuminating the artwork as the light bar passes underneath. The scanner sensor captures each of the color images in turn in a single pass. Figure 13.1 shows the interior layout of a typical color scanner.

Figure 13.1



This diagram shows how a color scanner works. The document is placed on the platen, as shown. A movable platform containing the exposing lamp, filters, and optical system is transported line-by-line underneath the document. In this example, a single lamp is shown. In such machines, three passes, each with a different filter, are made. You can also have three colored lamps, no filters, and a single scanning pass which captures all three colors.

The amount of light reflected by the artwork for each color varies according to the color of the pigments in the original. Cyan pigment, for example, absorbs red light and reflects blue and green. So, when a cyan area is illuminated by the red fluorescent light in a scanner, relatively little light is reflected. Most of the light produced by the blue and green fluorescents is reflected. Your scanner software records that area of the original as cyan. A similar process captures the other colors in your subject during the scan.

Some color scanners are flatbed models in which the light source moves under the subject. Others keep the light source fixed and move the platen itself. You'll find systems like this in larger, higher end flatbed scanners. Still other color scanners are hand models that you use by leaving the artwork in one position and moving the light source/sensor with your arm during the single pass.

At the very high end are still other scanning systems. As I noted earlier, the first color scanners were million-dollar devices used in the graphic arts industry; the artwork they scanned had to be wrapped around a drum. These scanners could simultaneously write a color separation on light-sensitive film wrapped around another section of the same drum, so that scanning and separating was a single step. Today it is common to scan first, then write to film in a later step.

Other high end color scanners use 2-D pixel arrays rather than linear scanners to capture an entire image at one time. These are very fast, very expensive, and limited in resolution because of the difficulty of making a sensor that can capture all the rows and columns of an image at once.

Color Calibration and Scanners

Color scanning offers new capabilities, but carries with it some added concerns. In most cases, you'll want your scanned image to closely approximate the original image. As we've seen, neither color display systems nor color hardcopy output devices are consistent nor particularly accurate. The best you can do is calibrate your peripherals so that there is a relationship between the colors in the original, what you see, and what you get as output.

The process is complicated by two facts. First, the response of any color system is rarely linear. And, to make things worse, it's difficult to describe a color in such a way that it means exactly the same thing to everyone.

Let me clarify these statements with a couple of examples. Assume for a moment a 24-bit system with 256 different tones of each color. A value of 0 for a particular color should represent no color; a value of 255 should represent the maximum intensity of that color. On a linear scale, 64 would represent about 25% intensity, 128 would be 50%, and so on. Yet, in real applications, an intensity of 64 is not half that of 128. It is some other percentage. The relationship of the actual representation to the ideal is known as a gamma curve.

Scanners do happen to conform to the ideal rather closely, but CRTs and printers tend to vary. If you know the gamma curve of a particular device, however, you can correct it. For example, if you know that, with a certain device, a value of 64 produces an intensity that is only 90% of what it should be to be linear, you can boost that value by an appropriate amount whenever it occurs.

This is done by building a gamma correction table that includes a value for each of the levels used in a system. The correction values can be automatically substituted by your software for the default values, theoretically producing a perfect, 45° gamma curve.

Now, all you need is a set of standards that can be used to determine what those correction values should be. A giant step in the right direction may be Eastman Kodak Company's PhotoYCC color interchange space specification.

The Kodak color management system is a set of tools and utilities that provides consistent color across devices and computer platforms. Kodak sells the system components to application/system software developers, CPU manufacturers, makers of displays, printers, and scanners, and to end users like you and me.

Among the industry leaders supporting or endorsing the new standard are Adobe Systems, Aldus, Autodesk, Next Computer, Olivetti, Phoenix Technologies, and Truevision. PhotoYCC is supported in PostScript Level 2.

PhotoYCC takes into account the differences between different color models; the calibration of printers, displays, and scanners; and the impact of external conditions such as fluorescent lighting. It is designed to be device and operating system independent and will work in DOS, Windows, Macintosh, Sun, and Unix 5.4 environments.

The system includes a developer's toolkit with a standard application programming interface (API) and a software color system processor. Users work within an environment that includes color management utilities and the color system processor. Device profiles measure and define the color behavior of a particular peripheral, while calibration tools enable them to perform up to the expectations of their profiles.

CRT screens are one of the devices that must be calibrated. Fortunately, as true color becomes more important, we're starting to see specific systems for doing such calibration. Color calibrators measure the output of your CRT at various color levels. Then, the system can adjust the levels of the electron beams in the monitor, if that is possible. Such systems require that each monitor have its own built-in sensors and calibration hardware. You can also use external hardware to make the measurements and then substitute new values for those in the gamma correction table used with that monitor. That's a cheaper solution, but it doesn't help when the colors produced by a monitor change because of age or user adjustments. Calibrating a CRT is shown in Figure 15.3

Radius, SuperMac Technologies, and other monitor manufacturers have introduced relatively low cost CRT color calibrators that allow you to calibrate their equipment easily. These were initially available only for the Macintosh.

Of course, getting true color on your screen is of little use unless the images you view bear some relation to your hardcopy output. Therefore, you must calibrate your system using the particular characteristics of your output device. Additional calibration steps are needed to match the colors produced by the printer with those seen on the screen. Printers have variations of their own, which are not limited to the exact hues of the pigments used. Dots may smear or enlarge as they are absorbed by the paper.

Because printers use the CYMK model, and CRTs use RGB, getting true color can be tricky, but, with enough care, satisfactory results can be had.

Some of the top software includes tools you can use to calibrate for your equipment. Picture Publisher, for example, has an extensive section in its installation guide on how you can calibrate the program for your particular equipment. Astral Development includes a 12-step gray scale, a measurement tool, and a calibration map dialog box within the software itself to help you make your adjustments.

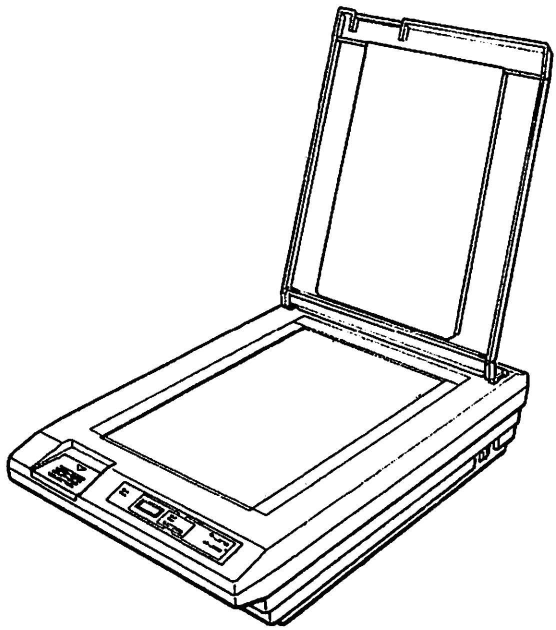
Gamma correction options provided by a typical color scanner will be discussed in the next section, in which we examine the features of the Epson ES-300C scanner in detail.

A Typical Color Scanner

In this section, I'll describe some of the features of a typical color scanner—specifically the Epson ES-300C that I use. It is currently the hottest color scanner on the market, displacing last year's darling, the Microtek MFS-300Z. All flatbed scanners introduced in the next few months will have similar features if they want to compete, so this description will give you a good idea of the capabilities color scanners will have in the future.

The Epson looks a lot like all the flatbed scanners you've seen over the past few years, except that it is quite a bit more compact. The unit is only 4 inches wider than its 8.5-inch scanning area, compared with an extra 8 inches for my HP ScanJet Plus. Thus it fits nicely on my desk between my Mac II and my Zeos 386 PC compatible. I've installed both the bidirectional parallel port for connection to the 386 and the optional SCSI interface that allows you to connect the scanner to the Macintosh. A slower serial port is also included in case you don't want to or can't use the other interfaces for some reason. The Epson scanner is shown in Figure 13.2

Figure 13.2



The Epson ES-300C color scanner

Front Panel Controls

The front panel has power on and ready LEDs, a display that shows the current reproduction ratio being used (and any error messages), a zoom/bright switch, and reproduction ratio switches. The latter two controls are not normally used when scanning. They are put to work when the scanner is connected to a color printer. In that mode, you can make color copies directly, and specify enlargement/reduction percentages with the front panel controls.

The Epson's glass document table measures 8.5 x 11.7 inches, including a very small non-scannable margin around the four edges, which varies from .08 inches (2 mm) at the right side to .24 inches (6 mm) at the bottom. This is a very small amount of image area to lose, and you can avoid even that by adjusting the original on the platen slightly so that all the image you want to capture is centered in the scannable area. The top cover is easily removable, so you can scan thick books and other unwieldy material.

Output resolution can be selected by your software from 19 different settings, ranging from 50 to 600 dots per inch. Everything over 300 dpi is interpolated from the 300 dpi information. You can specify the horizontal and vertical scanning resolution independently.

Single-Pass and Three-Pass Scanning Modes

As noted above, the Epson scanner offers a single-pass color mode, in which each line is scanned three times during a single pass. Three colored gas fluorescent lamps are used alternately. The scanner also has a three-pass sequence that scans the entire image in triplicate – once for each primary color. While the single-pass mode is faster, the three-pass method is thought to produce more accurate colors and a sharper scan, since there are fewer stops and starts within each pass. Frankly, I use the single-pass mode all the time. I've enlarged tiny sections of color photos that I've scanned and was able to see the fine grain (actually dye clumps; the silver grains are removed from color images during processing) of the original.

Color capability adds flexibility to the Epson scanner in its monochrome mode as well. You can select which of the three light sources (red, green, or blue) will be used for the monochrome scan, thus providing an optional drop-out color. For example, if red is used as the scanning light, all colors that contain green or blue are scanned darker and with more contrast, while colors with red are scanned lighter; only the proportion of the other two colors in the hue are recorded. Pure red is ignored entirely.

A similar phenomenon occurs with the other two additive primaries. So, if you have some artwork drawn on a blue grid, you can scan using the blue lamp to drop out the background grid. Or, if you are doing some OCR work with text that has been red-pencilled by the editor, you can drop out the red marks to avoid confusing the OCR software. Selection of drop-out colors can be used to improve the appearance of yellowed, stained, or otherwise defective originals.

Automatic Correction

The Epson scanner can automatically provide some image manipulation and correction for you. Three different halftoning methods can be chosen. It also has five gamma correction curves that can compensate for the idiosyncrasies of specific types of printers and displays. You can choose one gamma correction for 24-pin and laser printers and another for 9-pin and inkjet printers. A third set of corrections offers enhanced contrast and definition when you are scanning images that contain both pictures and text.

Connection Capability

The feature, other than the low price, that makes the Epson ES-300C so attractive to me is its connection flexibility. Because I can connect it to my IBM clone and my Macintosh simultaneously, it's like getting two scanners in one. Epson doesn't recommend this configuration, as it may cause excess RF interference, but I haven't had any problems. My office is separated from the rest of my house by a thick brick wall (my cordless phone won't even transmit through it), and my nearest neighbor is a quarter mile away. So far, no one's TV or VCR has complained about my setup.

Frequently, I have both computers fired up and scanning in the same session. I can scan an image with the Macintosh, then scan one on the IBM, and go back and forth as much as I want. I haven't noticed any problems with both computers trying to reinitialize the scanner, but then I usually turn it on before either of them is powered up (Macintoshes like to have all the devices on the SCSI bus turned on when they are activated).

If you have needs similar to mine, give the Epson a try and see if using it in a dual mode works for you, too.

Direct Printing

The second connection flexibility of the scanner is its capability of hooking up to a limited number of color printers for direct printing of scanned images. This allows you to have a color copier for a fraction of what you might expect to pay. At this writing, the scanner works with an Epson LQ-2500, LQ-2550, or LQ-860 printer with a color ribbon, an HP PaintJet color inkjet printer, or an HP 3630 color graphics printer with a parallel interface. The system won't work with a PaintJet equipped with an HP-IB or RS-232C interface.

If you're using an Epson 80-column printer or an HP PaintJet, the full scanning area of the ES-300C can't be printed the same size as the original. You must either reduce the image area somewhat or reduce the size of the final output.

For example, you can print an area measuring 7.1 x 9.3 inches at 180 dpi with an 80-column Epson printers or PaintJet and an area measuring 9.7 x 11.9 inches at 180 dpi with an Epson 136-column printer. That's at 100% magnification. If you want to reproduce the whole 8.5 x 11.7-inch area of the glass carriage, you must use an 80% zoom ratio with the 80-column printer or PaintJet.

You select the printer type you are using from a set of DIP switches that are easily accessible on the front of the scanner. You can then use the controls on the front to specify zoom level and brightness for the color copy you'll be printing. The zoom size range goes from 50% to 200% in 1% increments.

Keep in mind that color printers are much slower than scanners, so the scanner will stop during the scan to wait for the printer to catch up. A typical direct print takes about 7 minutes at 100% magnification and about 5 minutes at 50% magnification.

Software Galore

One of the best things about the Epson scanner is that it comes with everything you need to get started, including software. No fewer than three packages are furnished – ColorStudio, ImageStudio and a ScanDo desk accessory. One of these will surely meet your needs.

A variety of image editing programs can be used to control your scanner, but you must have a driver for your particular model. Epson doesn't just given you a collection of scanner drivers to use with software you purchase at additional cost; they give you the software, too. That heads off complaints that the still-new Epson scanner can't be used with popular packages like Adobe PhotoShop. If your favorite image editor isn't supported at this time, you probably received an equivalent program among the four that Epson packages with the ES-300C.

Summary

This chapter summarizes the demands placed on color scanners and explains how they work.

Color scanners use three different light sources—one each of red, green, and blue—to scan a color image. Some scanners make three passes over the image—one for each color. The Epson color scanner has three separate fluorescent tubes. The three alternate illuminating the artwork as the light bar passes underneath. The scanner sensor captures each of the color images in turn in a single pass.

The amount of light reflected by each color in the artwork varies with the color of the pigments in the original.

Some color scanners are flatbed models in which the light source moves under the subject. Others keep the light source fixed and move the platen. Still others are hand scanners for which the artwork remains fixed, and the light-source/sensor is moved by your arm during the single pass.

At the very high end are still other scanning systems. The first color scanners were million-dollar devices used in the graphic arts industry. The artwork scanned by these devices must be wrapped around a drum. These scanners simultaneously write a color separation on light-sensitive film wrapped around another section of the same drum, so that scanning and separating takes place in a single step.

Still other high end color scanners use 2-D pixel arrays rather than linear scanners to capture an entire image at once.

In most cases, you'll want your scanned image to closely approximate the original image, even though neither color display systems nor color hardcopy output devices are consistent nor particularly accurate. The best you can do is calibrate your peripherals so that there is a relationship between the colors in the original—between what you see and what you get.

However, the response of any color system is rarely linear. To make things worse, it's difficult to describe a color in such a way that it means exactly the same thing to everyone.

Scanners do happen to conform to the ideal rather closely, but CRTs and printers tend to vary. If you know the variation, or gamma curve, of a particular device, however, you can make the necessary changes to correct it.

Eastman Kodak Company's PhotoYCC color interchange space specification is a set of tools and utilities that provides consistent color across devices and computer platforms. Kodak sells the system components to application/system software developers, CPU manufacturers, makers of displays, printers, and scanners, and to end users like you and me.

CRT screens are among the devices that must be calibrated, and there are now color calibrators that measure the output of your CRT and help you make the necessary adjustments.

Getting true color on your screen is of little use unless the images you view have some relation to your hardcopy output. Therefore it is also necessary to calibrate your system using the characteristics of your output device. Since printers use the CYMK model, and CRTs use RGB, this can be tricky, but, with enough care, satisfactory results can be had.

The Epson ES-300C is the first of a new breed of low cost, flexible color scanners. It can be used with both IBM and Mac computers, even during the same session.

It offers a single-pass color mode, in which each line is scanned three times during a single pass. Three colored gas fluorescent lamps are used alternately. The scanner also has a three-pass sequence that scans the entire image in triplicate—once for each primary color. While the single-pass mode is faster, the three-pass method is thought to produce more accurate colors and a sharper scan.

Color capability adds flexibility to the Epson scanner in its monochrome mode as well. You can select which of the three light sources (red, green, or blue) will be used for the monochrome scan, thus providing an optional drop-out color.

The Epson scanner can automatically provide some image manipulation and correction. Any of three different halftoning methods can be chosen. It also offers five gamma correction curves that can compensate for the idiosyncrasies of specific types of printers and displays.

The scanner can also be hooked up to a limited number of color printers for direct printing of scanned images, with a full size copy taking about 7 minutes with a typical printer.

14

Color Separating and Your Scanner

This chapter, along with the last one, which introduced general color concepts, will be your introduction to desktop color separating concepts. It won't give you step-by-step instructions on how to make color separations. Rather, my goal is to explain the terminology and outline the key techniques to help you decide if you want to tackle this complex chore. After you've read the following material you'll be better prepared for the tutorials and instructions you'll need to do a good job.

Color separating is not a job for the desktop dilettante. You should be doing a fair amount of it to justify the education and investment it requires.

In fact, in my last scanner book, I summarily dismissed desktop color separating as not worth the bother. I stated with a fair amount of assurance that halftones and color separations were best left to traditional prepress procedures. That statement was about as well-received as Walter Mondale's platform plank calling for higher taxes.

Was I that far off base? After all, color printing is expensive, and two-thirds of the cost is typically incurred at the printer. The $700 price of a set of color separations using traditional methods can be reduced to about $150 with desktop procedures.

Yet, any saving you might realize by doing your own separations is apt to be a small part of the total budget for a given job, and the best quality is obtainable only with expensive, high end equipment, anyway.

I have since discovered that many of you are working hard to prove me wrong.

I blame my 15 years of writing about more traditional color separation techniques for my quickness to discount the possibility of doing it on the desktop. After all, during that span I've spoken to hundreds of color separators who use millions of dollars worth of equipment.

For example, in the late 1970's one of my regular assignments was to visit top newspapers like the Boston Globe to report on the winners of an annual newspaper color reproduction contest. Some of the top professionals in the country told me their secrets for achieving prize-winning color.

Recently, I visited a leading color separator in Madrid, Spain, who had the latest color pagination equipment and a hard disk "farm" that filled a room larger than my office. His shop prepared all the color for Spain's leading automotive and motorcycle magazines.

Instead of globe-trotting, I should have spent a little more time with desktop publishers who have neither the funds nor the inclination to work with such high end equipment.

No matter. Over the past few months, readers have been sending me samples of their work that prove just how wrong I was. One marketing executive sent a stunning full color brochure that was scanned, laid out, and color separated using nothing more than equipment embarrassingly similar my own. *Esprit,* a newspaper published by Douglas Ford Rea's students at the Rochester Institute of Technology, uses vivid color photography that is entirely electronically generated or scanned for desktop color separation. The folks at one software outfit are in the habit of sending me gorgeous, 6-foot posters created and color separated using their software.

Okay, just because I initially found color separating on the desktop somewhat daunting doesn't mean it can't be done. I plead creeping senility. This chapter makes amends with an overdue explanation of the color separation process—both the traditional and the new electronic versions. You can judge for yourself whether you want to get involved with color separating in your own work.

Traditional Color Separating Methods

You've already learned about the four standard process printing colors—the subtractive primaries cyan, magenta, yellow, and black. The printing press that will create the final output for your color separating efforts uses these colors on separate printing plates to create full color images. Printing can be accomplished either by running each sheet through the same press several times—each time with a new plate—or by a continuous run through a series of four presses arranged in a line.

With the latter setup, as with all offset printing, an arrangement of cylinders is used for each color being printed. The plate is fastened to one cylinder, which rotates to come in contact with a set of dampening rollers, which wet the plate so the non-printing area will repel ink. Then a set of inking rollers transfers ink to the printing area of the plate. The inked image is then transferred to a second cylinder, which has a rubber blanket wrapped around it to accept the ink. The paper passes between the blanket and a third cylinder, called the impression cylinder, to pick up the ink for the finished sheet. Some types of presses don't use impression cylinders but instead stack several printing units together so that both sides of the sheet can be printed at the same time. The blanket cylinder of one unit acts as the impression cylinder of the other, and vice versa.

The process by which the separations and plates are produced is known, quite logically, as the prepress process. The first step is to create four pieces of film that contain halftone images of the original in the proper proportions to print the cyan, magenta, yellow, and black colors.

Originally, this was done using camera techniques, much like those described in the chapter on making halftones. However, each separation halftone must be shot through a red, a green, or a blue filter to ensure that the tonal values are balanced for one of the primary colors.

Camera separations can be made using either direct or indirect screening methods. The latter is a two-step method (actually, two sets of steps); continuous tone (non-halftoned) corrected separation negatives are made, and then the actual halftone screening is done from them.

In direct screening, both the separations and the halftone screens are created in a single set of steps.

With either of these methods of color separating, color corrections are done with masks, which are low density copies of the original produced with special filters. Portions of the image are held back in certain areas of individual separations, subtracting color as desired.

In both cases, you end up with four pieces of film. All the separations needed to print a given color are combined on a master page in a process called stripping. Originally, this was a tedious hand process that required a great deal of care, because all the elements had to be taped up in perfect register. Finally, the pieces of stripped film for the individual pages are used to make printing plates.

Electronic Separating

Color scanners and electronic page layout systems can eliminate hand stripping by separating an entire page at once. You scan your images, assemble them as they appear in the finished piece, and then use the color separating features of your software to produce four files that can be sent to an imagesetter. The files include the registration marks the printer needs to align the films and plates perfectly for the printing step.

Generally, these are Encapsulated PostScript files, which are compatible with Linotronic and other imagesetters.

Color Proofing

One step that you will need to consider is the color proof. Even using desktop prepress systems to keep costs down, color printing remains expensive enough that it's a good idea to get a sample of what a page will look like before you run off a few thousand copies on an offset press. The press itself is usually not used for making the proof. When a printer's press is idle, the shop isn't making any money. So, you can't expect the printer to set up, print you a few copies for approval, and then sit there until you're ready to go.

Instead, another process is used to simulate a press proof. The goal is to get the colors of the proof to match the appearance of the press sheet. That's not always easy, however, because different pigments are used. Certainly, computer printers, even the $20,000 color models, won't do the job.

Most proofs are made using one of two popular proofing systems. One is 3M's Color Key, which consists of four acetate sheet overlays, each carrying one of the primary colors from your separation films. Laid over one another on a white sheet of paper, a $20 to $60 Color Key can give you a good idea of how successful your separation efforts have been. An advantage to Color Keys is that if your proof seems to have too much or too little of a certain color, you can re-separate that color and redo only that key.

DuPont's Chromalin system uses a single laminated sheet with all four colors to represent your color proof. While more accurate than Color Keys, Chromalins are also more expensive and must be remade from scratch each time you change even one of your color separations.

Selection of Screen Angles

We first looked at the concept of screen angles in the section on halftoning. You'll recall that black-and-white halftone screens are angled at 45° to avoid conflict with strong vertical and horizontal lines in an image.

However, you can't use the same angle for all four color separations, because the dots would overlap and obscure each other. Instead, a different screen angle is used for each—typically 75° for magenta, 105° for cyan, and 90° for yellow. If the correct angles are used, they create a rosette pattern.

The 45° angle is least obtrusive to the eye and is used for the strongest, most visible color—black. The magenta screen is angled at 75°, which is 30° from the black, and affords the least chance of moire effects between them. The cyan screen is placed 30° from the magenta at 105°. Yellow is placed at 90°—only 15° from the cyan and magenta—because yellow is the least visible color and any moire that results is likely to be insignificant.

Some Considerations

You must take into account that when you print one color inside the boundaries of a second color, a white gap may appear between the colored areas if the positioning is not exact. Typically, this is avoided in color separating by making one of the images slightly larger so it overlaps. You may be able to frame your image with a row of pixels in a transition color. This technique is called a trap and is provided automatically by some software; the lines of one of the separations are thickened, causing the overlap. (You can see that just making the images larger won't work; "holes" must become smaller at the same time that outside borders grow larger.)

Separating Line Art

Separating line art – whether scanned or created with a draw or paint program – is relatively easy. Programs like CorelDraw allow you to specify certain colors for specific sections of an image and then print those sections to files, which can be used to make color separations.

Separating Photos

Color photos are more difficult to separate, because you must deal with a range of color percentages rather than the fixed binary colors found in line art. Your separation program must take your scanned data, calculate percentages of colors, and transform them into a halftone pattern. Pre-Press Technologies' Spectre/Match Print and Publisher's Prism are two applications that can do this.

The former has a feature called global unsharp masking, which locates the boundaries between colors and increases the contrast between them to improve sharpness.

You can separate the photos you plan to use in your publication individually and then have them stripped in conventionally. Or, if you are feeling especially ambitious, you can separate an entire page, eliminating the stripping step. The latter technique provides greater control and even opens the possibility of creating files that can be used to make printing plates directly, eliminating the film step, too.

Color Correction

The discussions of color models that fill so many pages of this book were not included just to fill space. You can use your new-found knowledge of color to make color corrections as you scan and to help you make color separations from your scans.

Why do you need color correction? There are several reasons:

- To compensate for errors or deficiencies in the scanning process. Perhaps your scanner wasn't calibrated properly with the rest of your system. If not, you can correct slight color casts with your image editing and color separation software.

- The original photograph was imperfect. Inaccurate color can be found in both prints and transparencies. Most desktop scanners handle only reflective artwork, although a few have transparency attachments.

  A color cast in your original transparency has several possible causes. Daylight changes in color throughout the day, from slightly bluish at high noon to a more reddish tone in early morning and late afternoon. Transparency films accurately record those changes in color temperature. Daylight films exposed indoors and indoor-balanced films exposed outdoors without corrective filters will have reddish and bluish tones, respectively. You can sometimes correct for these casts, although in extreme cases nothing can be done to replace colors that are entirely missing from an original. Pumping blue into an extremely reddish photo produces neutral gray, rather than a properly balanced photo.

  Color negative films exposed under a variety of lighting conditions can be corrected successfully at the printing stage in most cases. If, however, a slight color cast remains on the print, it can be corrected in your image editing software.

- For creative reasons. You may want to change the colors in a subject simply to produce a rendering that will attract attention or make a creative statement.
- To correct for deficiencies in the printing process. You've already learned about dot gain as it applies to gray scale reproduction. Dots can enlarge during the color printing process as well. If you know how your press will behave, you can make corrections to compensate ahead of time.

  Another concern in the printing step is trapping. In this context, trapping refers to the ability of ink to adhere as well to another layer of ink as to bare paper. With poor trapping, some of the ink in some colors won't remain on the paper. You can correct for this ahead of time if you know about it—if not at the press, then by using techniques such as undercolor removal (explained in the next section) to reduce the amount of ink that must be overlapped to produce dark colors.

  A final problem in the printing process is the inability of any printer or press to reproduce the full range of tones found in an original photograph. As a result, the full tonal range that you can capture and manipulate with your scanner may be compressed on the final sheet. You can make corrections in your image before the printing plates are made to compress the colors in ways that are best suited for your particular subject.

  Understanding color theory can help you make corrections more easily. If your image is dark blue, for example, you should know that you must remove magenta and yellow to produce a brighter blue. If a red is muddy, it contains too much yellow and cyan and not enough magenta, so you must reduce the yellow and cyan and boost magenta. If a yellow is too orange, you must subtract magenta. These techniques will become second nature to you after a while.

Undercolor Removal and Gray Component Replacement

Undercolor removal is the removal of some of the cyan, magenta, and yellow from the dark shadow areas. This reduces the size of the halftone dots in each of these colors and reduces the amount of ink that prints in the shadows. Thus ink doesn't build up and pick off the sheets on the press. The black printer can then carry more of the detail in the shadow areas, improving quality and reducing cost by replacing expensive colored ink with cheap black ink.

Gray component replacement is a related process in which some or all of the three process colors are replaced by black ink over the entire image, not just in the shadows. This technique is used to adjust the overall density of an image, including the color areas, while undercolor removal adjusts only the neutral density or gray areas. UCR also improves trapping, as described above.

Two Stand-Alone Color Separation Tools

I have already mentioned that more image editing and page layout programs for the Macintosh are adding color separating tools, even when their capabilities are intended for separating line art and spot colors. You can gain additional flexibility and sophistication by turning to a stand-alone color separation package, which will give you greater control over the key parameters involved in making separations. Such software allows you to add color separation capabilities to your system regardless of whether or not your other software supports color separating. Two such packages are discussed in the following section.

Summary

Color separating on the desktop has become a realistic option for a growing number of PC users. They are finding that it is practical to lay out finished pages and produce color separation files for output to imagesetters.

Most desktop separations are destined for the offset press, which prints each sheet several times with inks colored with pigments in each of the three subtractive primaries, plus black.

As with all offset printing, the paper passes between a series of cylinders for each color being printed. The plate cylinder is dampened and inked, then transfers the ink to a cylinder which is wrapped with a rubber blanket that actually imprints the paper.

During the prepress process, the publisher or printer creates four pieces of film, which contain halftone images of the original in the proper proportions to print the cyan, magenta, yellow, and black colors.

Originally, this was done using camera techniques, which can produce separations using either direct or indirect screening methods. The latter is a two-step method; continuous tone corrected separation negatives are made, and the actual halftone screening is done from them. In direct screening, both the separations and the halftone screens are created in a single set of steps.

The separations needed to print a given color are combined on a master page in a process called stripping. Color scanners and electronic page layout systems can eliminate hand stripping by separating an entire page at one time.

You scan your images, assemble them as they appear in the finished piece, and then use the color separating features of your software to produce four files that can be sent to an imagesetter. The files include the registration marks the printer needs to align the films and plates perfectly for the printing step.

One step that you will need to consider is the color proof. Even using desktop prepress systems to keep costs down, color printing is still expensive enough that it's a good idea to get a sample of what a page will look like before you run off a few thousand copies on an offset press.

Most proofs are made using one of two popular proofing systems. One of these is 3M's Color Key, which consists of four acetate sheet overlays, each of which carries one of the primary colors from your separation films.

DuPont's Chromalin system uses a single laminated sheet with all four colors to represent your color proof. While more accurate, Chromalins are also more expensive and must be remade from scratch each time you change even one of your color separations.

Color halftones use a different screen angle for each color: typically 75° for magenta, 105° for cyan, and 90° for yellow. If the correct angles are used, they create a rosette pattern.

Separating line art—whether the art is scanned or created with a drawing or painting program — is relatively easy. Programs like Ventura Publisher allow you to specify certain colors for specific sections of an image and then print those sections to files that can be used to make color separations.

Color photos are more difficult to separate, because you must deal with a range of color percentages rather than the fixed binary colors found in line art. Your separation program must take your scanned data, calculate percentages of colors, and transform them into a halftone pattern.

You may need color correction to compensate for errors and deficiencies in the scanning process or if your original photograph was not perfect. You may also want to change colors for creative reasons or to allow for deficiencies in the printing process.

Undercolor removal and gray component replacement are two of the color correction tools you can use. Undercolor removal is the deletion of some of the cyan, magenta, and yellow from the dark shadow areas. This density is replaced by black, reducing the size of the halftone dots in each of these colors.

Gray component replacement is a related process in which some or all of the three process colors are replaced by black ink over the entire image, not just in the shadows. It adjusts the overall density of an image, including the color areas, while undercolor removal adjusts only the neutral density or gray areas.

15

Scanners and Output

This chapter deals with some of your output options, especially the high end output devices you will be using to print your quality halftones and color separations. The fast-changing world of PostScript and non-PostScript printers will be the focus.

The Complete Scanner Handbook has a chapter that deals in more detail with the advantages of PostScript and how that page description language works. If you're looking for more information, that's a good source.

Here, we review how printers work and then concentrate on some information not readily available elsewhere that you need to get the most from your scanned output hard copies.

It's no coincidence that many of the leading scanner manufacturers, including Epson, and Microtek, also offer printers. In fact, in the case of Epson, the companies are better known in the personal computer world for their printers than anything else. Printers are a vital link in the capture-to-output process, and scanner/printer vendors, like Vidal Sassoon, know that if you don't look good, they don't look good.

Most of what you scan ends up in one of two places: either as a disk file that is displayed electronically as part of a desktop presentation or in a publication of some sort. Publications far outnumber presentations and always will.

We are, however, probably many years away from the day when electronic direct-to-printing plate systems will take over the reproduction industry completely. Current laser plate burning systems are too slow; it's much faster to burn a plate from a piece of film. Therefore, desktop publications printed by offset presses will require some sort of hardcopy output that can be used to produce photographic film prior to platemaking.

That hardcopy can be output by an imagesetter or by some form of low resolution device, such as a laser page printer. Generally, you'll want to make hardcopy proofs on your own laser printer and send your files to a service bureau for final output. The proofing stage is not one that should be skipped, as it can cost you just a few cents to print a page on your laser printer but $1 to $10 or more to correct an error that has already been typeset.

If you don't have a laser printer, many service bureaus can print your files on their own and then either fax or messenger the proofs to you for final approval. Proofing can save you some expensive lessons. (I learned this hard way in producing my last book: I made some tiny corrections in a chapter and then sent the files to the service bureau without reprinting them one last time. I forgot to turn crop marks on when the file was printed, and the whole chapter had to be redone. It was such a blatant error that I would have caught it when the first proof sheet was printed had I taken the time.) This chapter will explore some of your other options.

Imagesetting Comes to the Desktop

It's great to be 43 years old. I was 16 when the Beatles hit their stride. I'm the same age as most of the leading movie directors and screenwriters, so when Rob Reiner and Stephen King set "Stand By Me" in 1960 when its characters were 13, that's when I was 13, too. I entered college when sweaters, ties, and fraternities were all the rage, and by the time I graduated, we were wearing blue jeans (and four of my classmates had been shot by the National Guard – it wasn't all great).

And I started working for my local newspaper at a time (1965) when they still used a couple of old Linotype machines to set classified ads in hot metal. Typesetting and I go way back.

That perspective makes the desktop revolution in typesetting all that much more amazing to me. When I started working at the newspaper, we reporters created our timely prose on manual typewriters, and the edited copy was rekeyed into a ponderous minicomputer system that set all the type photographically, using tiny sets of letters that exposed the image of each character onto photosensitive paper. These systems cost a lot of money but were still faster and more efficient than the Linotypes, because you didn't have to keep a metal mold for each size and font of type that you wanted to use.

About 25 years later, I was in the typesetting business myself, founding a tiny offshoot company called Laser Faire to serve the needs of local printers who wanted low cost, medium resolution typesetting overnight for simple work like company newsletters and business forms. That venture lasted less than a year, because in that time the local printers discovered that for a few thousand dollars they could equip themselves to do the same work and cut me out.

The ways in which we use typesetting have changed dramatically just in the time I've been writing books, too. My first book was submitted to the publisher in 1983 as a paper hard copy printed by the daisywheel printer attached to my Xerox 860 dedicated word processor. Someone rekeyed the whole thing for a typesetter.

My publishers began accepting, then requiring, book submissions on disk, and by 1987 they were inserting typesetting codes into the disk files themselves. In 1989, I handled my first project from start to finish: I now submit my books directly to a professional editor (the legendary Betsy Staples Ahl of Creative Computing fame), lay them out in Ventura Publisher, scan and create my own illustrations with packages like Canvas, proof on a laser printer, and wind up with PostScript files that are output by an imagesetter.

PostScript Makes It All Possible

You may find that the steps involved in your own work have been similarly telescoped. For in-house publications, the person who writes or edits the newsletter may also print out a master on a laser printer for offset or photocopy reproduction. Even more sophisticated jobs may remain with one person or a small team until they are finally sent to a typesetter. While you might have left the technicalities of type specification up to your printer a decade ago, you may now have to handle those details yourself. That's particularly true of scanner users, who are, by definition, involved with the production process in a hands-on way.

One of the reasons that the transition from desktop publication to professional typesetting has been so seamless is that PostScript has become a standard that crosses the boundary lines between platforms. Apple can take a lot of credit, along with Adobe, for introducing the first desktop PostScript printer, the original LaserWriter.

PostScript files are simple ASCII text files that have commands, which are carried out by a PostScript interpreter, installed somewhere in the job stream between the output file and the printer or imagesetter. It doesn't matter whether the PostScript file is created on a Macintosh or an IBM PC or a Unix system like Next. If the commands are correct, the file can be printed.

Nor does it matter what output device is driven by the interpreter. PostScript is device- and resolution-independent. If you send a text file containing 6-point type to a 300 dpi laser printer and to a 2500 lpi imagesetter, the raster image processor (RIP) connected to each translates the PostScript commands in a way appropriate for each output device. Such small type will appear rather coarse on the laser printer and a lot sharper on the imagesetter. Similarly, a scanned image printed to either will be printed within the constraints of their respective resolutions.

That makes it practical to create PostScript files on the desktop and proof them on a low cost printer (there are $500 inkjet printers with 300 dpi resolution which can be made PostScript-compatible). The exact same files you proof on your desktop can then be sent to a service bureau equipped with an imagesetter for higher resolution output. Of course, you might want to change the halftone screen resolution to take advantage of the imagesetter's smaller dots, but you don't have to.

When TrueType was announced by Microsoft and Apple, some were quick to pronounce PostScript dead. With TrueType slated for Macintosh System 7 and for OS/2's Presentation Manager and Windows 3.1

in the PC world, it seemed possible that the new standard would have a head start.

Since that time, the "if it ain't broke, don't fix it" syndrome has taken over, and millions of happy users of PostScript have indicated that they want to continue using a page description language that has worked very well for them. In the future, most applications will probably support both TrueType and PostScript, and TrueType/TrueImage printers, like the Apple Personal LaserWriter LS and Apple StyleWriter introduced in March, 1991, will have to be compatible with both page description languages. It's not even inconceivable that you'll be able to mix TrueType and PostScript on the same page.

Meanwhile, PostScript continues to grow and improve, even if it isn't broke. PostScript Level 2 has new halftone screening techniques, better memory management, speed improvements, JPEG image compression, and support for fonts that contain more than 256 characters. The latest version includes many of the functions that normally require coding by your application. So, your application can, say, draw a box by calling those functions instead of drawing the box itself.

In 1990, Adobe introduced its Emerald controller, a RISC-based PostScript controller that Adobe claims works three to seven times faster than previous controllers. The new controller has already been incorporated into offerings from Agfa Compugraphic, Autologic, Canon, Monotype, and Varityper.

You'll notice that we don't call them typesetters anymore. That's because they are no longer limited to type. If a PostScript file contains text columns, headlines, and halftoned images, an imagesetter can reproduce them all.

Before we get into the true imagesetters, let's look at your options for low resolution output of 300 dpi or less.

Low Resolution Output

It may seem strange to call 300 dpi devices "low resolution," but, like everything else, resolution is relative. Since most desktop publishers these days have access to true high resolution devices through service bureaus, 300 dpi really does fall somewhere in the middle of the scale.

For example, 300 dpi output is fine for proofing and may even be good enough for final copy if you're printing all the copies of your publication as originals. But once you start reproducing from 300 dpi masters, either through photocopying or by photographing laser originals for offset lithography, you lose some quality. I have produced quite a few newsletters from laser prints and find that, no matter how hard you try, text characters spread a little and scanned halftone images lose tonal range. In many cases, that loss of quality is acceptable. The results are much better than the dot matrix printing that was used by newsletter publishers in the past.

Your low resolution options break down into three different categories—dot matrix printers, inkjet printers, and page printers.

Dot Matrix Output

The first printers used with personal computers, like the daisywheel printers that followed, often produced fully formed characters. These were actually teletypewriters (frequently equipped with only uppercase letters) and were obviously incapable of any sort of graphics at all. The early dot matrix printers, like the 9-pin model I purchased for $900 in 1979 seemed a remarkable advance, because they could provide both upper- and lowercase characters and graphics of a low resolution sort.

In the decade-plus since, dot matrix printers reached an apparent peak in technology, around the time of, say, the ImageWriter II. Today, dot matrix printers are entrenched in a niche that may remain secure for another five years.

Dot matrix printers form images by hammering sets of tiny wires against a ribbon to place dots of ink on the paper. Those with 24 pins are capable of resolutions approaching 200 dpi and thus offer near letter quality (letter quality being the quality produced by nearly defunct typewriters and daisywheel printers).

A group of dot matrix printers in the $300 range enjoys popularity in offices that can't afford or don't need the better quality of higher priced inkjet and page printers. They use tractor feed or sheetfed paper, can take a minute to print out a single page, and generally provide years of trouble-free, low cost service. Until we see non-impact printers with higher resolution in the same $300 price range, I don't think dot matrix devices are in much danger of extinction.Scanner users may find them useful for proofing and even for final output when quality requirements aren't stringent.

Inkjet Output

Inkjet printers also use one-line-at-a-time techniques to reproduce images and offer resolutions as high as 300 dpi. They are often seen as the poor person's laser printer, because, if you can put up with speeds of less than a page a minute, you can frequently get almost the same image quality from a $500 inkjet printer as from a $999 laser device.

The new Apple StyleWriter, with its TrueType compatibility, 360 dpi quality, and $599 list price is a bargain most scanner users will want to consider. If one page a minute is fast enough for you, it makes an excellent device for proofing pages that will be sent to a typesetter. You can even use its output as originals for publications that will be photocopied.

The most common inkjet technology used today is called drop-on-demand in which dots are sprayed from a print head onto a sheet of paper. While special papers were required in the past for the best quality, the latest water-based inks adhere well to any paper.

Inkjet printers with three or more print heads can produce color output at reasonable speeds.

Page Printers

Laser printers have become so pervasive that the acronym NLQ has come to stand for near laser quality rather than the original near letter quality. However, much of the quality produced by these printers is attributable to the system used to apply toner to the paper at high resolutions. In practice, any type of mechanism can be used to write the individual pixels. And, in fact, liquid crystal shutters and light emitting diode arrays are frequently used in these printers. We just tend to think of them all as "laser" printers, regardless of how the mechanism actually works.

The writing mechanisms use a precise light source to image pixels from your bit-mapped page onto a photosensitive conductor, which may be a belt or a drum. From that point on, the process is more or less identical in all common 300 dpi page printers.

Printer Components

Page printers contain several components. These include the controller, a marking engine, a toning engine, and a mechanism for moving the paper from the supply tray to the delivery tray.

Controller

The controller may reside in the printer or in your computer. The computer downloads either a bit-mapped image or a set of PostScript instructions to the controller. The controller is sometimes called a raster-image processor (RIP) because a bit map is a type of raster image. The controller has access to the full complement of memory in your printer, some of which can be dedicated to storing bit-mapped images of various fonts.

The remaining available memory is allocated to the page images themselves. The more memory you have available, the more pages your printer can hold at one time, providing you with a type of printer buffer. Extra memory also expands the size and resolution options you have. That is, if you have only 512K of memory in your printer, you can print a graphic image no larger than a third of a page at 300 dpi. You may have to settle for 150 dpi resolution to image an entire graphics-filled page.

However, if you have 1.5Mb of memory or more and haven't consumed a lot of it with downloadable fonts, you can usually print a full page of high resolution graphics.

In PostScript printers, the controller has another function. It receives its printing instructions as a series of commands in the PostScript language. The built-in interpreter then constructs a bit-mapped page image using those instructions.

Marking Engine

Your printer can expose the image area, in which case it is called a write black printing engine, or it can expose the white areas on the page, skipping over those pixels that are to be printed black, in which case it is called, not surprisingly, a write white engine.

With write black systems, toner is attracted to those areas of the drum that have been illuminated by the writing light source. Such systems are better at defining very fine details, since only the pixels of the image are written to the photoconductor. All other areas of the drum or belt are left untouched so there is reduced chance of spurious pixels that can produce artifacts that reduce resolution. These printers are often preferred by those who use their laser output for offset masters, as they produce masters that are not too dark. These lighter masters reproduce better in offset systems in which ink tends to spread.

In write white systems, toner is attracted to the areas that are not illuminated by the writing light source. Because of the toning systems used, this usually results in much denser black areas. These systems are preferred for final output, because the black areas are denser and more consistent.

The Toning Engine

The rotating photoconductor belt or drum and the paper are given an electrical charge from a set of charging coronas. The toner is also electrically charged, either through the addition of charging agents or because of its natural electrical characteristics.

Toner contains the pigment and the somewhat larger carrier particles to which the pigment clings. The image areas on the drum are given one charge (positive or negative) while the surrounding areas have the opposite charge. Toner particles also have the opposite charge and are, therefore, attracted to the image areas and repelled by the non-image areas. The toner is picked up by the drum, which rotates and transfers it to the paper, which is charged to attract the toner. The paper then passes through a set of heated fuser rollers which permanently fuse the toner to the paper.

PostScript Options

Not too long ago, if you wanted PostScript output, you needed a PostScript printer, and PostScript printers cost anywhere from $3500 to $6000 or more. Today, the picture has changed dramatically. You can now output 300 dpi PostScript files for roughly $500, if you're willing to use a relatively slow inkjet printer and an add-on PostScript interpreter with only 13 fonts.

If that isn't acceptable, $1300 will buy you a laser printer that is up to ten times faster and offers "standard" 35 PostScript fonts first offered with the Apple LaserWriter. In the $2000 to $3000 price range, some truly fast and powerful PostScript machines are available.

A growing group of printers uses Adobe-licensed PostScript interpreters, while others use cloned versions of PostScript prepared from the specifications that Adobe released in 1990. However, PostScript can easily be added to printers that started life emulating ordinary PCL-based LaserJets. Everything I print is PostScript based, except for the output of the word processing program I am using to write this book, but I don't own a PostScript printer.

Instead, I use an add-on interpreter, either Freedom of Press or UltraScript, and print the files that I create with Ventura Publisher, PageMaker, Adobe Illustrator, and my other software. Even the longest jobs – including the 500-page proofs that make up a typical book – print just fine if you make the necessary allowances. In my case, that means using a backup computer to print the files, or letting my computer run all night. (This is one time when it's nice to have a printer with a 500-sheet paper capacity.)

This next section will show you how you can add PostScript to your non-PostScript printer.

Software Interpreters

You don't need a particular printer to use software-only PostScript add-ons like Freedom of Press or UltraScript. All you need is some free hard disk space.

Other similar products include PreScript and Freedom of Press. All of these will take PostScript files you print to disk from your application and convert them into a page your LaserJet Plus- or LaserJet II-compatible can handle.

Each of the software interpreters comes with its own selection of fonts. However, I've been able to use UltraScript and Freedom of Press with downloadable PostScript fonts from several sources with no problem. I have even used PostScript fonts I generated with Fontographer and other utilities.

These interpreters are fairly slow, taking 4 minutes or so to print a page that can be output by a genuine PostScript printer in less than a minute. All work best on a faster Macintosh. For pages without scanned images or other graphics you'll probably find them only a little slower than your printer's regular mode if you have a fast machine.

Medium to High Resolution Output

At some point, you'll want higher resolution output than a 300 dpi printer can provide. As you've seen, continuous tone images require higher resolutions to produce the combination of fine halftone screen frequencies and extended gray scales that you usually must have.

Once you get beyond 300 dpi, you're getting into the realm of the imagesetter. Costs start to rise sharply – usually to the $8000 level and up – which means that you can't dedicate the printer to a single user or allow it to sit idle for most of the day until you need its special features. Medium to high resolution is often the exclusive domain of service bureaus and corporations that have enough users – and uses – to justify the expense.

Imagesetters

An imagesetter consists of a PostScript raster image processor (RIP) and a writing engine that records the image onto resin-coated (RC) paper (for type or line art) or film (for halftones or color separations). Many are modular: you buy one box containing the actual recorder and a second with the RIP that suits your needs and budget. RIPs are actually powerful computers with PostScript interpreters built in.

Modular

Imagesetters can be expensive and hard to justify unless you have need of one for at least 8 hours a day. Monotype's Prism imagesetter will print at 1200, 1800, and 2400 dpi onto rolls of foot-wide film or paper. It costs $40,000. A Varityper VT600W costs $17,000 and comes with a $2500 annual service contract. If you don't do 15,000 pages a year for which you can charge someone else, you can't justify a 600 dpi device of that sort.

The most commonly used imagesetter is the Linotronic L300, which was one of the first machines that could be outfitted with a PostScript RIP. There are variations within the line, with a model number ending in 30 indicating that the unit has the powerful RIP 30 controller. The Lino L330 is especially well-suited for making color separations and boasts resolutions of up to 3386 dpi.

Other capable imagesetters are the Agfa Compugraphic 9400, Varityper 4300, Hell Graphics Ultra*Setter, BirmySetter 300, and Optronix ColorSetter 2000.

The choices within a single vendor's product line can be bewildering. For example, within Agfa's modular ProSet line, you can choose either the 9400 or 9800 model, equipped with either a PS Max RIP or the faster PS Max Plus RIP. Agfa also markets the SelectSet 5000 for color work. It includes the high speed Adobe Emerald controller I mentioned earlier and can handle tabloid-sized pages. Agfa also offers a lower cost StudioSet 2000.

Given the price tags on these units, representatives of the vendors will be delighted to help you through the maze of models to find the imagesetter that is best for you.

While the price of these units has dropped from $250,000 down to $50,000 or less, remember that you must also have at least two people trained to use them (in case one of them gets sick or is on vacation) and, maybe, a second unit to use when the first is down for maintenance. You may be able to save the cost of the second unit by making an agreement with another company or a service bureau to handle your overflow work and emergencies.

Imagesetters must be carefully maintained and calibrated daily, and the film, chemicals, and laser must be monitored for quality. As you can see, imagesetting is not something you do to save time or money. It is best approached as a business and given your undivided attention. If your commitment is anything short of that, you're better off using a service bureau, even if your volume is high enough that it might seem to cost you a little more in the long run.

Using a Service Bureau

Service bureaus are a convenient alternative for most of us who do only a few pages of typesetting a week. You generally don't even have to leave your office to use a service bureau, because courier services will deliver your file disks overnight to any bureau in the country. The firm I happen to use is located on the West Coast, and I use Federal Express to send them six or seven disks at a time.

If I have only a few pages to be set, I upload the files directly to the bureau's computer bulletin board. In some cases, I have been able to finish a job, upload it to my service bureau 3000 miles away, and receive finished output by 10:30 the next morning. In this case, it happens to be an advantage to use a West Coast bureau. If I finish a job at 5:00 p.m. Eastern time, they receive it about 2:00 p.m. their time. The coastal time warp can add an extra three hours to my workday.

Service bureau rates are based on an average time to print a page—say, 3 to 10 minutes. If you include many scanned images in your pages, the pages can take much longer, in which case you may be charged an extra fee of $50 to $100 per hour for those more complex pages.

Advantages of a Service Bureau

There are a number of advantages to using an outside service bureau.

Sufficient Control

You don't need to follow a job all the way from start to finish to have sufficient control to guarantee the quality you want. Because the same files can be printed on your laser printer and your service bureau's imagesetter, you can proof in-house, then send the files out to the service bureau for final output. Let them worry about maintaining their imagesetter and other equipment to the standards you and their other customers require.

Quality

Because they do so much work, service bureaus are likely to do a better, more accurate job than you could do, even factoring in your personal tender loving care. Sheer volume makes them good. In addition, the materials used in imagesetters—resin-coated paper and film—are more dimensionally stable than the paper used in higher resolution, plain paper typesetters.

Imagesetters are also rugged, more precisely engineered pieces of equipment (they cost more, too), so even at equivalent resolutions, a service bureau's equipment will offer sharper images, better registration, and reduced banding (one type of which is caused by variations in transport speed) than plain paper typesetters.

As you know, registration is critically important in color work. If you don't think so, talk to veteran users who insist that all the separations in a set come off the imagesetter consecutively. Exposing one or more separations at different times on the same machine can cause problems in some high quality applications.

Technical Support

A service bureau may have an operator who can read PostScript well enough to troubleshoot jobs that run into snags while printing. You'll receive your job back when you expected it, rather than an apology.

They'll know how to handle wide pages and closely registered colors and be able to talk you through any problems you may have generating your PostScript files in the first place.

Speed and Capacity

Regardless of how good an inexpensive alternative to an imagesetter might be, it's unlikely to be as fast as a good imagesetter. Can you turn out 300 pages of typeset material in a day? Probably not. Is it likely you'll never, ever need to churn out that much imagesetting? Don't bet on it.

New Color Options on the Horizon

I'm going to conclude this chapter with a discussion of some of the latest innovations in color output. This is an overview to let you know what's coming down the road.

For scanner users, color printers are an extremely convenient proofing device. None of them are cheap enough or fast enough to produce final output in full color—unless you are producing an extremely short run publication (under 100 copies or pages).

Color printers today use one of several technologies. A few are glorified photocopiers, with three or four of toning stations to apply colored toner powder to paper that has been electrostatically charged three different times—once for each of the color images.

Others use three inkjet print heads to lay down the primary colors in fine dot sprays or three-color wax ribbons to transfer the colors through a thermal process. These are all expensive, fairly slow, and limited in resolution because of the need to dither images to generate a full range of colors.

Printers like the Kodak XL 7700 use varying amounts of dye to produce the full 16.7 million color gamut at full resolution.

Mead Corporation has invented a new technology that will be used first in color copiers but which could find its way into color printers in a few years. Mead's Cycolor copiers use a special polyester film coated with billions of light-sensitive microcapsules called cyliths that contain the dyes in the three primary printing colors, mixed with light-sensitive cyanine borate salts.

When this film is exposed to light, it produces a negative image of the original. The color capsules that are not exposed are hardened and deactivated. Then the film negative and the final paper sheet are squeezed together through a set of rollers. The remaining capsules burst, transferring the ink to the paper, which is then heated to fuse the color image.

With a laser, light-emitting diode (LED), or liquid crystal exposing device, this technology could easily be applied to color printers for Pcs.

Summary

Printers are a vital link in the capture-to-output process, since most of what you scan ends up in one of two places: either as a disk file that is displayed electronically as part of a desktop presentation or in a publication of some sort.

Desktop publications printed by offset presses require some sort of hardcopy output from an imagesetter or other device that can be used to produce a photographic film prior to platemaking.

One of the reasons that the transition from desktop publication to professional typesetting has been so seamless is that PostScript has become a standard that crosses the boundary lines between platforms.

PostScript files are simple ASCII text files that contain commands that are carried out by a PostScript interpreter. It doesn't matter what platform is used to create the PostScript files, nor does it matter what output device is driven by the interpreter. PostScript is device- and resolution-independent. If you send a text file containing 6-point type to a 300 dpi laser printer and to a 2500 lpi imagesetter, the raster image processor (RIP) connected to each will translate the PostScript commands in a way appropriate for each output device.

When TrueType was announced by Microsoft and Apple, some were quick to pronounce PostScript dead. Since that time, millions of happy users of PostScript have indicated that they want to continue using it.

In the future, most applications in will support both TrueType and PostScript.

Your low resolution options break down into three categories—dot matrix printers, inkjet printers, and page printers.

Most of us use page printers, either for final output or for proofing. Much of the quality these printers offer is attributable to the system used to apply toner to the paper at high resolutions. Any type of mechanism can be used to write the individual pixels, including liquid crystal shutters and light emitting diode arrays.

The components of a page printer include a controller, a marking engine, a toning engine, and a mechanism for moving the paper from the supply tray to the delivery tray.

A growing group of printers uses Adobe-licensed PostScript interpreters, while others use cloned versions of PostScript prepared from the specifications that Adobe released in 1990. However, PostScript can easily be added to printers that started life emulating ordinary PCL-based LaserJets, using add-on cartridges and interpreters that run in your computer.

Imagesetters are high resolution devices that aren't limited to type. If a PostScript file contains text columns, headlines, and halftoned images, an imagesetter can reproduce them all.

Imagesetters can be expensive and hard to justify unless you have need of one for at least 8 hours a day. While the price of one of these units has dropped from $250,000 down to $50,000 or less, remember that you must also have a backup unit and at least two people trained to use them.

Service bureaus are a convenient alternative for those who do only a few pages of typesetting a week. You generally don't even have to leave your office to use a service bureau, because courier services will deliver your disk files overnight to any bureau in the country. You can also telecommunicate files by modem.

Service bureaus offer better control, higher quality, speed, and technical support at a reasonable price.

One final output concern of many scanner users is color output. Color printers are convenient proofing devices.

16

Scanners And Operating Environments

You might not think of your Macintosh operating environment, also known as your system software, as part of your scanner toolkit, but your choice of configuration can have a dramatic effect on what you can and cannot do with your scanner. For example your system software can:

- Limit or expand the amount of memory available for your scanning and image editing applications.
- Provide caching to speed up disk-intensive operations.
- Allow you to work with several scanner applications at one time, so you can jump back and forth between a scanner controlling program, an image editing package, and a page layout system at the click of the mouse.
- Make your working life easier or more difficult, depending on how you have your system configured.

On first look, your choices as a Macintosh user might seem to be fairly limited. You can use a recent version of System 6.0 (say, System 6.04 to 6.08), depending on the requirements of your applications (some work only with specific versions or later). You can elect to activate MultiFinder, or not. In May 1991, another option became available: System 7, which eliminated one choice by making MultiFinder active all the time. Of course, there is always AUX, if you happen to need the UNIX operating system and its applications for some reason. (UNIX is beyond the scope of this book, so that's the last you'll hear of AUX.)

If you examine the operating system/system software question more closely, though, you'll see that there are many more options available than you might have thought, just within the two major Mac system software versions. This chapter examines some of them and shows you how to select a configuration to enhance your scanning.

Some History

History is the story of people, not just a dry listing of facts, so I hope you don't mind if I inject some personal experiences into this background on operating systems.

My first personal computer didn't have an operating system to speak of, at least not in terms of what we think of today as an operating system. That's because my very first desktop computer didn't have a disk drive. There was no boot-up process to speak of, either. When you turned the computer on, the CRT instantly displayed the message **Memory Size?**, which let you type in a figure to reserve some memory for machine language programs. If you just pressed Enter, the system displayed copyright information, the statement **READY**, and a > prompt. Those of you familiar with the IBM world will recognize that as a C> prompt, but without the C drive letter; there wasn't a disk drive, remember? The prompt indicated that the computer had loaded BASIC and was ready to accept BASIC command line statements (like NEW and LIST) or a series of numbered program lines. BASIC, was, in effect, my operating system, and, in that first computer, let me do anything that could be accomplished using a very limited set of Tiny BASIC commands and 4K of memory.

Believe it or not, we were very happy with this breakthrough back in the late 1970's. Earlier computers had required loading some sort of supervising program through paper tape storage devices, or even by manually throwing a set of switches on the front panel. I happened to escape such horrors, because I waited for a cassette-based system before I took the plunge with my first microcomputer.

Finally, an Operating System

That was in 1977. Within a year, I got one of the early floppy disk drives and started working with one of a series of disk operating systems. The reason I needed a DOS was that computers require a software overseer to manage anything more than a limited set of peripherals. My first computer worked fine when the system consisted of a keyboard, CRT, and some memory. Once the industry started moving toward more sophisticated types of peripherals, BASIC was no longer satisfactory as an operating environment.

Of course, each software program that you run could easily communicate directly with all the components of your computer—from the microprocessor itself to the CRT screen, keyboard, printer, and hard disk. That would be an extraordinary waste of human and computer resources, however. Each time a programmer began developing software for a personal computer, he or she would have to develop routines to handle common chores, such as input/output to the disk and writing to the screen.

It's very likely that a programmer would actually develop such code only once and then re-use it over and over within a program or even among several programs. Even so, there would be a great deal of duplication of effort, because each programmer would in turn be re-inventing the wheel. The programs themselves would be longer than necessary because many of them would incorporate routines that duplicated those found in other programs.

A program interface can be written to fit between the hardware and the software applications. When a program wants to fetch information from the hard disk, for example, it needn't interrogate the disk to see how it happens to store information or how that information is cataloged. Instead, the request for data can be sent to the software supervisor, which accesses the hard disk and forwards the information—or perhaps an explanation of why it couldn't be found—to the program. All the software a user might have can use this method, allowing the common software interface to replace individual program modules.

With the Macintosh, Apple carried this concept to the logical next step. Nearly all the routines that interface with your hardware are present either in ROM (as with QuickDraw), or loaded into RAM from disk (as with 32-bit Color QuickDraw in systems like my Mac II that don't have it built-in.) Your software doesn't need to concern itself with how to draw anything on the screen, nor what type of screen you have. It can call the QuickDraw routines as required, and rely on your System to keep track of the resolution and size or your display.

This is in stark contrast to the IBM world, in which software must have special drivers for eac h type of display and printer that is installed (*separate* drivers for monitor and output device!). These drivers cause incompatibilities, rob memory, and prevent users from matching their favorite applications with new hardware that may not yet be fully supported with drivers.

Advantages of an Operating System

The simpler arrangement in the Macintosh world makes the most of the strengths of an underlying operating system. There are several advantages to this scheme over and above simplifying software applications. The supervising program can be changed, if necessary, by the manufacturer of the computer to compensate for significant differences in hardware. The commands received from the applications programs remain the same; the only thing that needs to change is the way the interface program puts them to work with a specific hardware configuration. That's why you can run many types of 24-bit color applications in 8-bit color mode. The System software handles the differences in how images are displayed.

If you want a demonstration of how important System software can be, just look at the changes that System 7 brought to the Mac world. It's fair to say that most of us who switched gained brand new computers, with some exciting new capabilities. In most cases, the software applications we run under System 7 remain the same. The new Finder and System capabilities give every program you run new flexibility.

Let's look at some of the ways in which your operating system can affect you as a scanner user, in a little more detail.

- Limit or expand the amount of memory available for your for your scanning and image editing applications. System 6.x won't let you use more than 8 mb of memory without a special utility. System 7 has the capability of using virtual memory on Mac IIx, IIcx, SE/30 and Mac II's with a PMMU chip, and allows up to a gigabyte of actual RAM for computers with 32 bit adddressing (the Mac IIci, IIsi and IIfx at this writing).
- Provide caching to speed up disk-intensive operations. Both System 6.x and System 7 have disk caching built-in. You should allow 32K of RAM for a disk cache for every 1 megabyte of RAM in your system. That's 64K for a 2 megabyte Mac, and 256K for an 8 Mb system.
- Allow you to work with several scanner applications at one time, so you can jump back and forth between a scanner controlling program, an image editing package, and a page layout system at the click of the mouse. You could do this with System 6.x if you were running Multi-Finder. System 7 makes MultiFinder active all the time, so more users will take advantage of this feature.
- Make your working life easier or more difficult, depending on how you have your system configured. There are many examples of how the System software can make your work easier. For instance, with System 7 you can set up alias icons to group applications and data files together, even if the actual programs and files reside in different folders.

  There's no need to hunt around on your hard disk (although the new Find feature makes it a lot easier to search for files when you do need to locate them. New hard disk management tools in System 7 also streamline many operations. For example, you can see hierarchical outline-format views of nested files and folders; it's not necessary to open each folder in a window to see what it contains.

  There's a lot to like in System 7, and I expect that many scanner users will be switching in the near future. Only those having computers with just a single megabyte of RAM will tend to lag behind. As we've seen, scanner users tend to favor more RAM and hard disk storage, and can derive genuine benefits from the System 7 improvements.

Let me point out just one feature that will be of tremendous use for scanner users: the alias feature. One use for alias that you might not have thought of us to greatly expand your hard disk space. You can create aliases for little-used scanned images or even programs. Store the originals on floppy disks. Keep the aliases in a convenient location on your hard disk, within a well-organized nest of folders.

You'll then be able to browse through these folders searching for scanned images or applications. The Mac's long file-naming capabilities makes it easy to apply descriptive names to the aliases. The aliases actually take up very little space on your hard disk. When you double-click on one, System 7 will prompt you with the name of the disk you should insert to access the original file. The alias linkages provide you with a simple, automatic, offline filing system for your scanned images and other files.

Summary

This chapter on operating environments for the Macintosh was designed to get you thinking about one of the hot topics of the year: System 6.x versus System 7 software. Contrary to what you might think, your Mac's operating system can affect you as a scanner user. Your choice of system has a bearing on how you use your scanner and its applications. The speed at which your computer operates, the amount of available real RAM and virtual RAM, and a number of convenience factors all are dependent on the operating system. System 7 provides some real benefits for scanner users, including the ability to access more memory, and more convenient ways to have several programs loaded at once, under the latest version of MultiFinder.

17

Other Tools and Tips

I'm still a long way from exhausting the vast reservoir of tools and tips that can help you do better work with your scanner. (As *Byte* Magazine columnist Jerry Pournelle would put it, there's still a pile of stuff on my desk that I just couldn't get to.)

To bridge the gap, this chapter provides a few odds-and-ends hints for getting the most from your past and present scanners, beginning with a useful technique for scanning from color transparencies. You'll also find a description of an experimental method for capturing color scans with a gray-scale scanner.

Scanning Color Slides with a Flatbed Scanner

Microtek has introduced a low-cost (street price under $2500) color slide scanner, the ScanMaker 1850, that should become *very* popular. Other scanners, particularly the more expensive models with color capabilities, have attachments that also let you scan color slides.

Such add-ons are not a must, however, especially if your need for scanning slides is modest. You can scan from color or black-and-white transparencies, overheads, and similar material fairly well with any flatbed scanner—if you know how. You may have to settle for black-and-white output, even if you have a color scanner, unless you're willing to experiment with light sources and some other parameters. This section will give you step-by-step instructions that will help you get started with this interesting technique.

Constraints

Flatbed scanning of color slides may sound like the greatest thing since radial tires. Before diving into this technique, however, you should be aware of some of its limitations.

- **You may not always get top-notch results.** Keep in mind that a 35 mm transparency measures only 1 x 1.5 inches (24 x 36 mm). At 300 dots per inch (dpi), that's only 135,000 pixels of information. If you scanned a more typical 4 x 4-inch piece of reflective art at 100 dpi, you would capture more detail (160,000 pixels, in fact).

  Even so, I've been surprised at the quality I've been able to obtain with this technique. Given a high-quality original and a small reproduction size, you should get eminently usable images. If you are scanning from 120 format (2.25-inch roll film) transparencies or overheads, your results will be even better.

- **The technique is time-consuming.** Scanning slides is not nearly as convenient as scanning reflective art—at least, not when you are using a flatbed scanner that isn't meant for such work. Simply preparing and positioning each slide takes a lot longer than scanning a similar number of reflective artwork originals. So, don't plan on making color slide scanning a high-production part of your scanner repertoire.

- **You can't capture the color in color slides by using a gray-scale scanner.** Of course! I just wanted to remind you. *You'd* never make a mistake like that, but there are people who would. I have actually overheard consumers trying to decide whether to buy color or black-and-white videotape, and I recently read of a desktop publisher who sent a gray-scale PostScript file to a service bureau and asked that it be output on a color printer.

- **You may not even be able to capture the full color range in your color slides with a color scanner.** I'll explain why later in the chapter.

Some Technical Background

Most of the scanners described in this book use reflective lighting to capture images from opaque artwork or documents. As you learned earlier, the scanner's light source is bounced off the original artwork or document, focused through a lens, and captured by a sensor within the scanner.

why scan transparencies

In contrast, transparencies are viewed by transmitted light, so the light source in the scanner won't do you much good. Not enough light passes through the transparency and is reflected back through it by the platen cover to record an image. Therefore, you need to replace the scanner's light source with your own, which backlights the transparency to provide the required transillumination.

Luckily, that's entirely feasible. With a strong enough backlight source, your scanner's sensor will ignore its built-in illumination and capture an image of the transparency. The trick is providing that source (I'll show you how shortly).

Merely backlighting the transparency isn't enough; the characteristics of the light source can have an effect on the image you capture. Here are some key properties to consider:

Intensity. The amount of light your scanner's sensor receives in a reflective system is stable, and confined to a relatively small range of intensities. The light source is, of course, the scanner's built-in lamp, which remains at more or less the same output level throughout its life. The amount of light reflected from your artwork is also limited, because even the whitest substrates can bounce back only, say, 80 to 90 percent of the light they receive.

In contrast, a backlit piece of artwork can flood the sensor with much higher levels of light. Clear portions of the transparency may let nearly all of the light source pass through, and that source may be much brighter than your scanner's built-in lamp.

So, you must select your light source or position it to provide a reasonable amount of illumination that won't overwhelm your scanner's sensor. But that can be dicey. Transparencies can represent a much greater range of tones than reflected art; "whites" can be so much whiter, while many dark tones can still hold details.

Contrast. When light is passing through a transparency, it can have contrast characteristics of its own. A tightly focused point-source light, such as a bare high-intensity lamp bulb, will be inherently much more "contrasty" than, for instance, a white fluorescent tube. The point-source light will simultaneously tend to enhance both details and defects in the transparency.

Diffused light sources, on the other hand, will reduce the contrast of a transparency, while hiding details and diminishing any defects in the slide. You can convert a point-source lamp to a a diffused source by passing the light through one or more layers of plain white (preferably ungrained) paper or cloth.

Color balance. All light sources aren't the same color—they can vary widely. A lamp that appears white to your eyes may actually be quite reddish. Our brains compensate for us; without some less biased means of comparison, we never know the difference.

Color scanners are less forgiving of these differences. A video scanner (camcorder) must be "white balanced" to compensate for the actual color balance of the ambient illumination. Video shot indoors under the daylight setting will appear too red; outdoor scenes that have been balanced for indoor light will seem excessively blue. That's because incandescent indoor lights are actually much redder than daylight.

Scientists, photographers, and others who care about color balance refer to the *color temperature* of light as a measurement of how reddish or bluish it appears compared with a standard, such as daylight. The color temperature is the actual temperature of a theoretical object called a black body radiator. When heated to roughly 5,000 to 6,000 kelvins, this object will emit light that approximates daylight. Household lamps produce a light equivalent to a much lower temperature, from 2,800 to 3,200 K. If you think of "red-hot" (really hot) and "white-hot" (even hotter), you'll see how a 3,200 K light can be classified as reddish, while a 5,000 K light is considered bluer.

For color reproduction, 5,000 K is the standard light source used in graphics shops to judge color separations. Actually, more than the color temperature affects how the color looks. Some light sources, particularly fluorescent lights, don't have equal amounts of each color in the spectrum. There may be gaps that affect the *color rendering index.* For our purposes, however, this won't be a concern. After all, you won't be able to produce separation-quality scans with the flatbed scanning technique. At best, you'll be able to capture color images that will be useful for inserting in desktop presentations created with programs such as Microsoft Powerpoint.

For less critical applications, it's enough for you to know that if you want to capture color images from a color transparency, the light source must be balanced close to 5,000 K. Otherwise, you'll need to compensate by adjusting your scanner's color controls.

How to Scan Slides

For easier and more repeatable slide scanning, it's a good idea to make a slide holder. Begin with a piece of cardboard the approximate width and length of your flatbed scanner's glass platen. You'll be able to position this board in approximately the same location on your scanner time after time by sliding it up against the edge guides or stops in one corner.

Cut a hole measuring about 1.5 x 1.5 inches in the cardboard, if you'll be scanning only 35 mm transparencies. Prepare other holders to accommodate slides of any other size that you'll be scanning.

Attach the slide to the holder. You can simply tape the 2 x 2-inch slide mount to the holder. The emulsion side of the film should face up. Which is the emulsion side, you say? One side of the transparency should be shinier than the other; that's the film base. The other side contains the actual image. If you view the slide at an oblique angle, you may be able to see the image in bas-relief. The slide will also be cupped slightly toward the emulsion side if the film isn't mounted between slips of glass, as is the case with some custom-mounted slides.

Clean the slide, if necessary. Because of the very large magnifications you'll be using, even tiny dust particles will be greatly magnified. You can use a soft brush or compressed air to remove dust. Don't try to use liquid cleaning solutions to remove more stubborn dust or fingerprints unless you know what you are doing, because you can easily damage the slide.

Place the holder face down on the scanner platen, so that the emulsion of the film is closest to the glass. Light that passes through the image and then through the base layer of the film will be scattered or diffused slightly, degrading the image. If you place the transparency's emulsion side closest to the sensor, no diffusion occurs.

Moreover, this configuration puts the image as close as possible to the platen, in approximately the same focal plane as a piece of reflective art. The film will be raised off the platen only by the thickness of the slide mount. With a thin plastic or paper slide mount, the extra thickness won't matter.

If your slides are in thick glass mounts, or if you want the ultimate quality possible with this technique, you may want to remove the film from the mount and tape it directly to the holder. Although this makes the process even more time-consuming, it may be necessary. Photo stores sell lint-free, disposable gloves that you can use to handle the film without leaving fingerprints.

I usually place two small but thick books on either side of the slide holder to ensure that the cardboard is wedged as tightly as possible against the glass platen. On top of the books, roughly 2 inches above the slide, I put a piece of plain white paper. A caveat: Don't use heavily grained or watermarked paper, because the pattern can show through in the finished scan. A piece of milk-white plastic or Plexiglas™ is ideal, but not essential.

Your light source should illuminate the diffuser paper or plastic sheet. For black-and-white scans of color or black-and-white slides, I use an ordinary high-intensity gooseneck lamp.

You may need to experiment with a few scans to determine where to place the light. The farther away from the diffuser, the lower the intensity. Find a setting that matches your scanner's sensitivity.

If your scanner software lets you perform a Preview scan, do so to locate the exact area of your slide opening in the holder. If you adjust the scan margins so that they are slightly larger than the slide area, you can perform rescans with those settings, even if the holder isn't positioned in exactly the same location each time. Make sure you don't scan too large an area. At the resolutions you'll be working with, scanning very large areas will chew up a lot of real estate on your hard disk drive.

Indeed, when you're scanning a 35 mm slide, it's almost always a good idea to use the highest true (noninterpolated) scanning resolution your unit allows—anywhere from 300 to 600 dpi. As I pointed out earlier in the chapter, even at the highest resolutions, there is only so much detail that you can capture from a 1 x 1.5-inch slide.

If you're scanning a 120 format or larger transparency, you may able to use lower resolutions and still get good results. Figure 17.1 shows a cross-section view of how slide scanning is done.

About Hand Scanners?

Can you use the slide scanning technique with hand scanners? Sure! After all, the 2.5-inch to 4-inch scanning width of a hand scanner is closer to that of a transparency than is the scanning width of a flatbed scanner. In fact, if you take a few precautions, a hand scanner can actually make scanning slides a little easier.

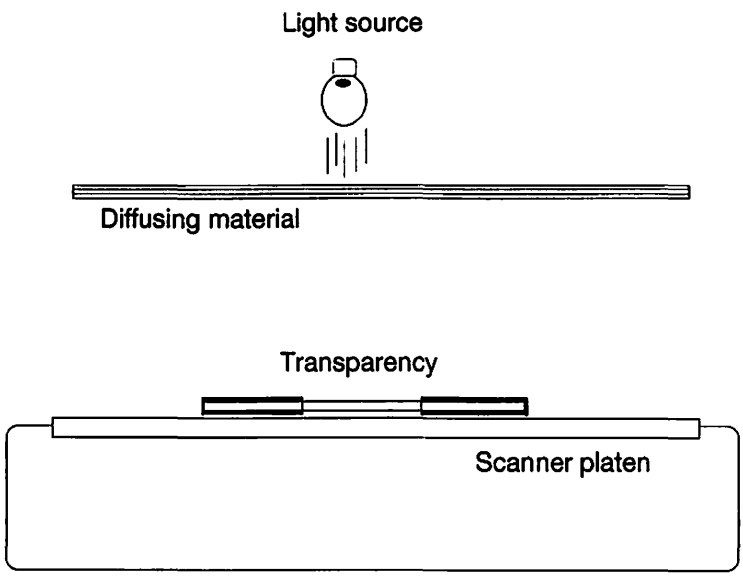
First, though, you'll need to turn the system upside down. Place the slide face up on a *light box*—a sandwich made of glass and white diffusing material, with a light source underneath. Then pass the hand scanner over the transparency. Take care that the scanner doesn't touch the surface of the slide. You may be able to get the sensor somewhat closer to the image than when using a flatbed scanner, so your quality is potentially equal or better.

Scanning Color

As I said earlier, if you want to scan in color with a color scanner, you'll need to experiment with various color balance settings to compensate for the excessive redness of most incandescent color sources. Usually, jiggling the blue control up and down until the color appears normal will do the trick.

Or, your light source can closely match the balance your scanner is set for, usually 5,000 K. Some types of fluorescent lights match this specification and have a spectrum that is continuous enough not to cause problems with serious "gaps." Check with your local photo store for a book that lists the color balance of various fluorescent lights by brand name.

Figure 17.1



Scanning slides with a flatbed scanner

Color Scanning with a Gray-Scale Scanner

Here is another experimental technique for those of you who have more time than money, or who just like to explore new ways of using the equipment you already have.

You may not get top-notch results by making color scans with a gray-scale scanner, but it *is* possible. You just have to know what to do.

Constraints

As with scanning transparencies, you'll find that there are a few limitations: Using a piece of equipment to do work it was never designed for definitely puts a crimp in what you can and can't do. But if know where the potential roadblocks lie, you can swerve around them to minimize or even avoid disappointment.

Artwork, especially line art, that has only a few colors scans much better than do continuous-tone color originals. Exact registration can be less crucial, especially if your original contains only primary colors. In that case, this technique will do quite a good job of separating the individual colors.

- Proper registration is a must. You must work very carefully. Consequently, the procedure can be extremely slow.
- You may have to experiment to find the right color filters that will do the job for you. A lot of fine-tuning may be necessary.
- Some gray-scale scanners won't be able to compensate for the reduced illumination that results from filtering the light source. Your images may be excessively dark, and you may have trouble correcting them with your editing software.
- You'll need a color image editing program that can work with individual color channels, such as Adobe Photoshop, to combine the finished images.

Technical Background

The technique of using a gray-scale scanner with color slides takes advantage of the fact that color scanners are actually little more than gray-scale scanners with three light sources or filters, one for each of the primary colors. Color images are scanned either three times in succession through each of the individual filters, or in one pass by using alternating light sources for each line. You can duplicate the former procedure simply by inserting a color filter between the original and your light source from one scan to the next.

Not all gray-scale scanners will work for color slilde scanning. For starters, you need a white light source that can be filtered to provide the proper balance of red, green, or blue light. That leaves out most hand scanners, because the yellow-green or red LEDs that they use for illumination consume less power. (Color hand scanners use a different type of light source.)

Also, hand scanners don't allow the precision of scanning required for good registration of the separate images. Some of the earliest color hand scanners were used in a fixed position; you placed the scanner's window on the area to be captured and then activated the scan.

Most flatbed scanners use white illumination that does permit color scanning. The problem is finding a way to put a filter between the subject and light source from scan to scan without moving the original.

How to Do It

First, you need to obtain the correct color filters. Any old red, green, and blue filters won't do; in fact, some of the "magic" in making camera separations comes from the printer's preference for this particular red, or that particular green. Filter selection can help compensate for some of the shortcomings of your original artwork, whether you are using a process camera or an electronic scanner.

Traditionally, camera separations have been made with a set of filters identified by Kodak Wratten numbers. Usually a Wratten No. 25 red (or sometimes a No. 29 red) filter is coupled with a Wratten No. 47 blue, or No. 47B blue and No. 58 green filter. You can purchase these filters in sheets from graphic arts supply houses.

An alternative is to use filters intended for theatrical lighting. Rosco, the supplier of choice for motion picture and stage productions, has offices in New York and Hollywood. If you can't find a theatrical supply house near you, order large sheets of filter material directly from Rosco; see Appendix C for addresses. The Rosco red filter best suited for color scanning is No. 17, the best green is No. 86A, and the best blue is No. 63.

Next, you must find some way to position the original so that it remains in the same place for all three scans. Actually, it's the angle of the image with respect to the scanning sensor that matters, more so than the x and y coordinate positions. If the image is moved a little to the left or right, or up or down between scans, you can compensate by moving around the individual layer in your editing software until it is positioned in perfect register with the other layers. That won't work, though, if the image is rotated even a tiny bit. A fraction of a degree's movement makes it impossible to register the images perfectly.

That's why it's best to keep the artwork fixed in relation to the scanning sensor for all three scans. One way to do this is to tape the original to the lid of your flatbed scanner, with the image side toward the glass (of course). If you use this method, you can carefully lift the lid between scans, slide out the old filter, and insert the new one.

This method works best with scanners that don't take an automatic document feeder (ADF), and thus have a platen cover that cannot be removed easily. Scanners that do accommodate an ADF often have lids that you can remove just by tugging on them. Their mounting systems may enough slack for the lid to change position, depending on how you close it.

Another option is to tape the artwork directly to the glass platen. Apply tape only to the top and bottom of the art, leaving the sides open. You should be able to slide a filter underneath, especially if you cut the filters a little large. Be careful not to disturb the artwork as you change filters. Radio Shack and other electronics stores sell a special type of removable tape for labeling videotapes that has just enough adhesive to hold most artwork. You can also use Post-It™ notes as semisticky tape.

If your scanner permits adjustment of brightness during the scan, set your unit for the highest allowable value. Otherwise, your scans are likely to be fairly dark. As you scan with each filter, save the image under a name that will make it simple to keep track of which is which, such as CATHY1.RED, CATHY1.GREEN, CATHY1.BLUE, and so on.

The final step is to merge the individual color scans. Some editing software packages enable loading separate images and merging them into individual color layers or channels. With Adobe Photoshop, for example, you can load red, green, and blue channels, and then combine them with the Merge Channels selection on the Mode menu. Specify RGB color when prompted. You'll probably need to adjust color and brightness to optimize your image.

And that's all there is to it!

101 Things to Do with a Dead Scanner

The only thing more certain than technological progress is that it is most likely to happen immediately after you've purchased the hardware of your dreams. As soon as you get your new scanner out of the box, installed, and working, someone comes along with a model that does more at a lower list price. When this happens, what do you do with your suddenly obsolete scanner?

Actually, veteran computer buyers aren't bothered by planned (or unplanned) obsolescence. As industry sage Doug Clapp once pointed out, "The problem with waiting to buy a computer is that, while you're waiting, you don't have a computer." In other words, buy what you need to do your job *now*. The scanner that will debut at the next Comdex may have a lot of bells and whistles, but it won't get your current jobs done when you need them.

For example, just before starting this book, I laid out new money to buy an interface card for an old, obsolete Hewlett-Packard ScanJet. Then I mated the ScanJet with a brand new computer. Why did I do this? Because the ScanJet does a specific job for about half the cost of newer products.

Keep in mind that new equipment is often announced a few weeks to a few months before it is actually available for purchase. Then it takes several more months before the product pipeline is full enough to allow widespread discounting among retailers and mail order firms. As a result, you probably won't be able to buy that impressive new piece of gear you read about this week at a decent price for four to six months!

But if you hook up Old Faithful to a new computer, you'll get good scans out of a reliable, proven product, as I did with my ScanJet. That's what I usually do. In fact, if I'm using anything that happens to be state-of-the-art technology, I know that (1) it was probably sent to me on loan from the manufacturer; (2) I may be one of the lucky few to discover a serious problem with it; and (3) in six months, it will represent trailing-edge technology that is still marvelously useful.

So, obsolescence does not always arrive as soon as you think. But when it does, how to upgrade and what to do with your old scanner may be of concern. Here are some tips to help you ease the pain as you figure out how to get a few more miles out of a not-quite-dead scanner.

- **Don't throw the baby out with the bathwater.** You may not have to start from scratch when you upgrade. For example, interface boards from an older model scanner may be compatible with the newer version. When Hewlett-Packard upgraded the original ScanJet to the latest ScanJet Plus model, the company added some major enhancements. For example, the new model enabled capturing 256 rather than 16 tones of gray, the orientation on the copy board was reversed, and so on. But the interface board was essentially unchanged, just a bit more compact. In a similar situation, you may be able to upgrade by buying only the scanner, thus saving $400 or more on a new interface board. The best way to tell if this is a practical option is to call the manufacturer.

  What do you do with your old, interfaceless scanner? In most cases, the old scanner and its interface board probably couldn't be sold for much more than you would have had to pay for a new interface board alone. That is, you might have cleared $500 for the pair and then wound up paying $400 for a new interface that was more or less identical to the one you just sold. You're better off foregoing the extra $100 and keeping your old scanner as a backup. How else could you get an extra scanner for $100?

You might also find another scanner owner with an "orphaned" interface board. My original ScanJet was used with an IBM PS/2 Model 60 and had a Micro Channel Architecture (MCA) adapter card. To move it over to the Macintosh, I just needed a new interface. Your old scanner and my old adapter card could be a marriage made in heaven for someone who uses an MCA-bus machine.

- **Use the old scanner for OCR jobs.** My original ScanJet can capture only 16 tones of gray, so I use it with my backup computer today primarily for OCR work. OCR software doesn't care how many gray tones you can capture, as long as your scanner has enough latitude to image both light and dark text pages. Resolutions higher than 300 dpi are useless for OCR jobs, because the extra detail isn't needed for differentiating characters and only slows down processing. Some OCR software actually limits the type sizes that can be read at higher resolutions.

  You might even want to upgrade your old scanner, purchasing an ADF that makes the equipment useful for OCR jobs. Text capture can be accomplished more or less automatically, leaving your new scanner free for graphics and color work.

- **Use the old scanner for line art.** Again, the number of gray levels or the lack of ability to capture color is not a consideration when you are capturing line art. When I am scanning logos, small line art graphics, and similar items, I use an older scanner.

- **Give your old scanner to a business associate, your church newsletter staff, a youth group, or a worthy charity.** This makes a lot of sense if you used your old scanner in business, have fully depreciated or capitalized it, and would just have to pay taxes on any salvage value you recovered by selling it. You can't take a tax deduction, but you can get a lot of satisfaction from seeing your scanner put to good use.

  A business associate in another location might be able to use your old scanner to capture images and text that can be telecommunicated to you when you need them. You never know when this capability might be a lifesaver!

A church group or other organization that is producing a simple newsletter with desktop publishing software can dress up its publication with scanned images. Your scanner may not serve you well any longer, but it could be a dream come true for a nonprofit group with no scanner at all.

Some of my ancestors on my mother's side were Native Americans who practiced a conservation philosophy that would serve us well today. For example, if an Indian planned to soften a piece of deerskin by soaking it in water, he or she would bury it in the soil beneath a parched tree. In watering the deerskin, the Indian also provided moisture for the tree. Similarly, making a small gift of a scanner donation today could pay off for you in the future.

Summary

This chapter provided a few hints for getting the most from your present—and previous—scanners, including a technique for scanning from transparencies.

Scanning a color transparency, particularly in black-and-white, is not that difficult. You need to make a slide holder that will enable placing the emulsion side of the slide as close as possible to the sensor. You can then backlight the transparency with a diffuse light source so that your scanner can capture the image.

You may be able to use a color scanner to obtain color scans of color slides if you have uses for the image other than making color separations. It's also possible to make color scans with a gray-scale scanner. To do it right, you need a set of red, green, and blue filters, as well as a way to keep the subject matter in register between scans.

This chapter also offered some advice on what you can do to head off unplanned obsolescence:

- You may be able to recycle your old interface board when you buy a new scanner.

- You can use your old scanner for OCR jobs.
- You can give your old scanner to a church newsletter staff, a youth group, or a worthy charity or nonprofit organization.
- You can give your old scanner to a business associate in another location for using in capturing images and text that can be telecommunicated to you when you need them.

18

Using the Software Diskette

Book and software combinations are not new. In fact, my second computer book, published in 1983, included 20 games, graphics programs, and applications on a cassette tape, along with program listings and programming tips.

Not long after that, diskettes started appearing in computer publications. The first that I can remember was the demonstration version of Microsoft Word bound into the October 1983 issue of *PC World* magazine.

Book/diskette combos really hit their stride when the enormously popular DOS Power Tools by Paul Somerson (aided by other contributors to *PC Magazine*) headed for the half-million sales mark. The Macintosh world wasn't far behind, with books such as *Stupid Mac Tricks* hitting the best-seller list in 1991.

Today, book/diskette combos offer a remarkable value for readers if the software is any good and the book itself well written. For about $30 to $35—only about $10 more than the price of a book alone—you get a diskette packed with useful software. In some cases, there are no further strings. In others, the software is provided on a shareware basis, and you're asked to register your copy with the original author if you find it useful.

Everyone benefits. You get demo programs, shareware, and freeware, along with well-written, comprehensive documentation, for only a small initial outlay of cash. Commercial vendors and shareware authors get broader exposure for their products. And authors, such as myself, can boost the value of their efforts with excellent software offerings.

I'm especially excited about the software provided with *The Complete Scanner Toolkit, Macintosh Edition.* Purchasers of this book receive four superior applications: Studio/1 1.0, QuickGIF, PCXtc, and GrayView.

Studio/1 is a combination monochrome painting and animation program from Electronic Arts that lets you create moving images from scanned files and from those you modify. The program on the diskette accompanying this book is a demonstration version. All Studio/1 features and capabilities are intact, except for the ability to print or save the animations that you create.

If you upgrade to the full commercial version of Studio/1, you get a 300-page manual, a slide show utility, and a driver that can link your animations to HyperCard stacks. The full program, which readers of this book can purchase at a substantial discount ($55 instead of the $79 list price) also has a selection of demonstration animations, templates, and a HyperCard stack that you can learn from. You'll find more information on Studio/1 in Chapter 19; instructions for ordering a copy of the commercial software appear in Appendix D.

The Graphics Interchange Format (GIF), invented by CompuServe, has become a popular graphics format for scanned images, because it allows the same files to be downloaded and viewed on Macs, IBM PCs, and other platforms. Many programs can save files in this format. All you need to view them is a program such as QuickGIF, included on the diskette. QuickGIF can open and view GIF, PICT, and MacPaint files, and is notable for its speed—roughly two to four times faster than previously available utilities.

You can also save files in either GIF or PICT format, so this program offers an easy way to convert PICT files to GIF and vice versa.

QuickGIF is a shareware program, which means that it is a fully functional, copyrighted application. You must pay for it if you use it, just as you would for any commercial software. However, you get to try the software before you buy it. This eliminates the risk of paying for a package, only to discover that it doesn't do what you want.

The copy of QuickGIF included with this book is a complete version that you may evaluate for a reasonable period. If you decide not to use it, you can pass along copies freely for your friends and associates to use under the same restrictions. You get a free nonshareware program, QuickGIF Plus, when you register your copy of QuickGIF with the program's author.

Appendix B explains the shareware concept in more detail. I provide a brief introduction to QuickGIF in Chapter 21. Appendix D gives registration instructions.

The third program on the diskette is PCXtc ("PC Ecstasy") which is a translator/viewer for files created in the PCX format. It allows Macintosh users to look at PCX files, which many scanning programs for the IBM PC create. If you have a screen capture program such as SnapJot, you can grab the images that you view and then save them to disk for editing with your favorite image program.

The PCXtc version included with this book is version .8. Because PCXtc is a freeware program, you do not have to register your copy and there are no restrictions on how you distribute it. The program is copyrighted, however, so you cannot alter it.

A later version of PCXtc, version 1.6, is a commercial (not shareware or freeware) program that you can buy from its author for only $10; see Appendix D for purchasing instructions. Version 1.6 adds the ability to save PCX files in MacPaint format and to print them directly, without the need for an intermediate utility such as SnapJot.

The last application on the enclosed diskette is GrayView, which is an interesting gray scale editor that was originally going to be a commercial product, but which fell by the wayside when the author decided to concentrate his efforts on completing graduate studies in mathematics. After you've seen this program at work, you may hope that David Fry tries his hand at Mac programming again in the future.

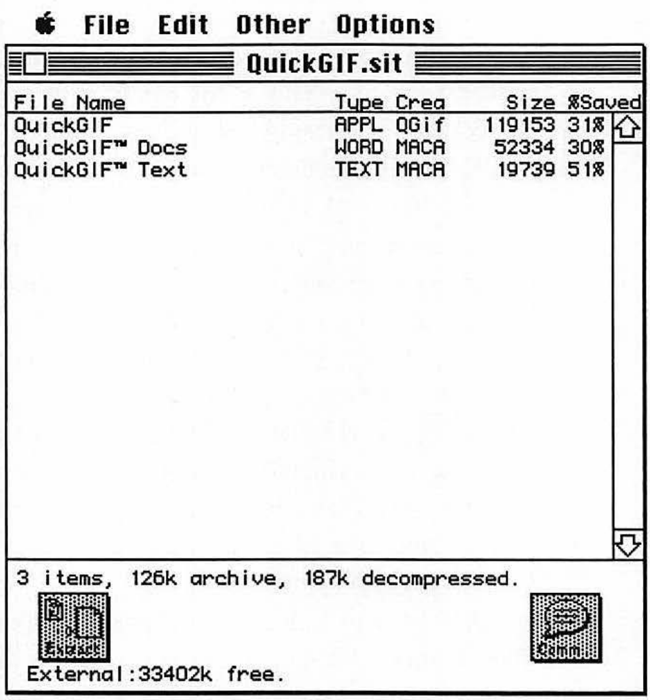
GrayView was originally written to convert Thunderscan files to 32-tone true gray files. However, it will also manipulate MacPaint, PICT, and some TIFF files. Not all its features are implemented, but I think you'll enjoy playing with it.

The disk also includes several practice files in TIF, PCX and MacPaint formats, which you can use to test out the enclosed software. The image is that of an abandoned 12th Century castle located between Toledo and Avila, Spain.

Unstuffing the Diskette

The four applications and image archive are provided in compressed form on an 800 KB floppy disk, plus the freeware utility UNSTUFFIT 1.5.1. Although there is some room left on the diskette, there isn't enough for all three of the applications in their uncompressed form. Therefore, you will need to copy the files to your hard disk, or copy the individual Stuffit archives (they each have an .SIT extension) to individual floppy disks. You use the application UNSTUFFIT 1.5.1 to extract the files from each archive. UNSTUFFIT itself is a valuable utiity that you can use to uncompress files that you download from bulletin board systems or on-line information services such as CompuServe. UNSTUFFIT is shown in Figure 18.1

Figure 18.1



UNSTUFFIT 1.5.1 allows uncompressing archived files.

Once you have copied the files and put your original diskette in a safe place, launch UNSTUFFIT by

double-clicking on its icon. Pull down the File menu and choose Open Archive. Find the .SIT file that you want to unstuff, and click on its name in the file selector.

UNSTUFFIT extracts both individual files and folders, which can contain more than one file or even other folders. If more than one file or folder is in the archive, you can choose to uncompress them one at a time, or you can uncompress just a single file. UNSTUFFIT puts the files and folders in the folder that you designate. It's a good idea to create a separate Studio/1, QuickGIF, and PCXtc folder for each application.

That's all you need to know to extract the files on the diskette. For instructions on using the individual programs, see Chapters 19. 20 and 21.

Summary

This chapter described the three applications and utility included with *The Complete Scanner Toolkit, Macintosh Edition:*

- Studio/1 is a demo version of a powerful monochrome painting/animation program.
- QuickGIF lets you view GIF image files, and is notable for its speed.
- PCXtc lets Macintosh scanner users view PCX files that have been created on IBM PCs and compatibles. (For information on how to transport such files to the Mac, review Chapter 9.)
- UNSTUFFIT 1.5.1 is a de-archiving utility that can extract all three of the applications from their compressed formats.
- GrayView is a gray scale editing program with some special features.

19

Animation and Studio/1

Scanners are a natural tool for the animation programs that add movement to the Macintosh screen. You can scan a line drawing or a gray-scale or color image, and then bring it to life with packages such as MacroMind Director (at the high end of the cost spectrum) or Studio/1 1.0 (at the more affordable end).

Animation should be one of the fastest-growing scanner areas for a simple reason: If a picture is worth a thousand words, a moving picture can communicate 10 times as much information in the same amount of time. An exploded drawing doesn't have the impact of an explosion; a set of pictures showing how to assemble a component is less valuable than watching the parts actually being fitted together.

This chapter looks at animation in general and how scanners fit in, and then helps you get started with the demonstration version of Electronic Arts' Studio/1 1.0 software.

Some Background

It takes at least a fraction of a second—if not longer—for the nerve impulses that result from an image focused on the retina of a human eye to fade away. That's why you can stare at something bright and then see an image of it when you turn away or close your eyes.

Motion pictures and video are possible because of this phenomenon. You view individual frames at very high rates of speed, usually 24 frames per second for movies and 30 frames per second for video. Your eye blends one image into the next, giving the illusion that you are viewing a continuous image. Because each frame differs slightly from the last—a figure may gradually move from the left side of the frame to the right—our eyes and brains merge all the still images to provide a sense of motion.

When the individual frames are captured in real time with a motion picture or video camera, the result is very lifelike. In fact, "Star Wars" effects wizard Doug Trumbull has developed a system called ShowScan, which uses images captured at 60 frames per second. Many viewers find it hard to tell ShowScan-generated images apart from reality.

Yet we don't need to produce quite that level of realism to communicate our messages. Frames assembled one at a time from a series of drawings, paintings, or movements of inanimate objects provide an effect that we call *animation.*

This sort of moving image can be lifelike, depending on the accuracy of the artwork. It can also be decidedly unlifelike, representing images that exist only in the imagination, from Gertie the Dinosaur (the first animated cartoon, from the early 1900s) to mutant adolescent turtle martial artists.

Computer CRTs present information in rapidly changing frames. The challenge in producing real-time computer animation lies in updating the Mac's video memory quickly enough to provide a sufficiently high frame turnover rate to generate the illusion of animation.

It's not that the video memory or display isn't fast enough. But the information that makes up each frame must come from somewhere, and that's usually a hard disk drive. You just can't read information from a hard disk quickly enough to paint 20 to 30 different images on a screen each second. And, as you've learned, full color or gray-scale images require a lot of disk space. Imagine what would be needed to store the 200 or so images in a 10-second animated clip!

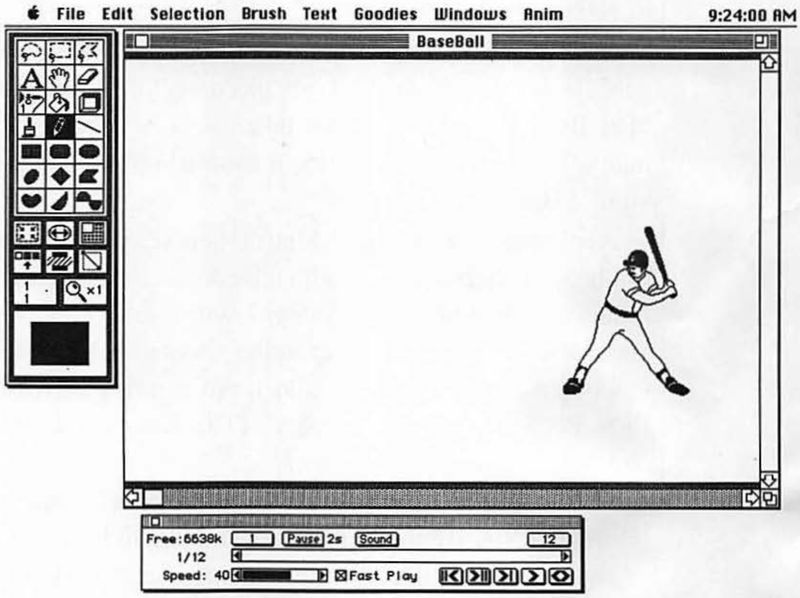
Fortunately, there are ways around the limitations I've described. One technique is to reduce the size of images. A 320 x 200-pixel image demands only 20 percent as much processing power and data transfer speed as a 640 x 480-pixel image. Another method is to use less pixel depth. Instead of 24-bit or even 8-bit color, we can get by with 16 or fewer tones.

The latter technique is the secret of Studio/1. This program uses only binary, MacPaint-type images. That brings down the processing overhead and storage requirements substantially, to the point where many animation sequences can be stored in fast silicon memory. However, you're limited to animating line art and dithered gray-scale images.

Those of you who have a Mac Plus or Mac Classic, or who want to get your feet wet in animation at a very low cost, may not find this much of a limitation. Studio/1 gives you a way to experiment, but still lets you produce effective presentations. The program is shown in FIgure 19.1.

After using the demo program, you may want to upgrade to the full commercial version for $55 (regularly $79), or investigate a more advanced animation product. Either way, I think you'll be impressed

Figure 19.1



Studio/1 from Electronic Arts enables any Macintosh owner to explore the fun of animation, and to create useful "movies" that can be incorporated into HyperCard stacks for business reports and presentations.

with the version supplied with this book. The only thing you can't do with this otherwise fully functional program is save your animations to disk or to print files. In other words, you can create presentations as lively and action filled as those produced with the complete version of the program, but if you want to actually use them for anything, you'll have to cough up the modest fee to buy the real thing.

What Is Studio/1?

Studio/1 is an extremely sophisticated monochrome painting program. If it had been available seven years ago, it would have blown MacPaint out of the water. True, you can work only with black or white pixels, as you do in MacPaint. With Studio/1, however, you aren't limited to 72 (dots per inch) dpi resolution. You can edit images at up to 300 dpi.

Studio/1 also provides a PostScript-compatible text layer and editable Bezier curves, such as advanced vector-oriented programs. The "Fat Bits" concept has been taken to a new level: There are eight magnification ratios, including a zoomed-out mode that lets you view all of a large image at once.

Apple Scanner users can control their scanners directly from within Studio/1. Others can scan with a Desk Accessory such as Scan/Do, or, if they use MultiFinder or System 7, with a scanner program loaded into another layer. Users can import the scanned images either through the Clipboard or directly, as Studio/1 can handle PICT, MacPaint, TIFF, EPS, PICS, and S1AN formats. (The latter is Electronic Arts' own compressed animation format.)

All the flexible selection tools and the resizing and distortion capabilities that you'd expect are there. You can add the gradients (dithered, of course) that are useful for producing backgrounds and other effects in presentations. These features make Studio/1 an excellent paint program for presentations or just fooling around.

But wait—there's more! Studio/1 lets you *animate* the images you draw or scan. You don't have to tediously redo each in-between image, either. You can specify a start point and an end point for an object, as well as the number of frames the motion should cover. Studio/1 generates all the images between the two points.

There are also special effects for transitions, fades, and distortions. Electronic Arts includes templates for animation, but you can also create your own. I especially like Studio/1's ability to link a specific digitized sound effect to a frame, allowing you to produce the sound of a baseball popping into a glove, glass breaking, and other appropriate noises. If you have a utility such as Sound Mover, or if you can use ResEdit to cut and paste sound resources between applications, this added dimension can be almost as much fun as the animation itself.

Make no mistake, Studio/1 is a powerful tool. You can use it with HyperCard to produce demos and demonstrations. And the Gallery utility included with the complete program lets you animate your own slide shows.

All in all, I've had a lot of fun with this program. That's why I wanted to share it with you.

Using Studio/1

Studio/1 doesn't demand the most advanced hardware. A Mac Plus or Classic with two floppy disk drives will work fine; a floppy disk drive and a hard disk drive are even better. You can use any of the upscale Macs, from the SE to IIfx, too. System 6.0.2 or later software is a must.

Once you've unstuffed the Studio/1 files from the disk included with this book, you need to put them in a separate folder, either on your hard disk or on a floppy disk that you'll use to run the program. As mentioned earlier, the demo version provides all the Studio/1 functions. except for Print and Save. It's worth noting, however, that the complete version of the program also includes a HyperCard driver and demo stack, a full disk of sample animations, and the Gallery slide show animation utility.

As with most Mac programs, you really don't need a manual to get started with Studio/1 (although the commercial version of the program comes with an excellent 300-page manual, more than you'd expect with a $79 program!). You can use the Help... option on the Apple menu to learn about Studio/1's painting toolbox, animation control panel, and the most frequently used keyboard shortcuts.

The best way to get acquainted with Studio/1's animation features is to run through these three Quick Start exercises.

Quick Start 1: Creating a Flock of Flying Birds

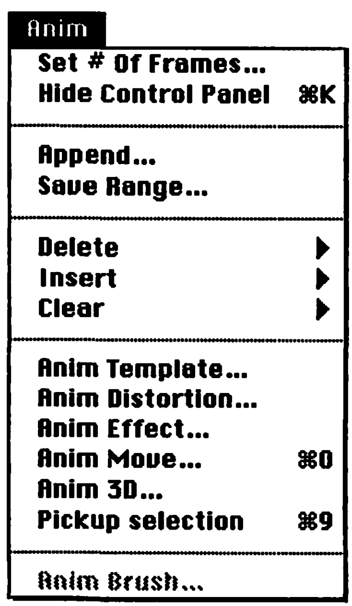
This exercise will show you how to use Studio/1's animated brushes ("animbrush") capability. An animbrush is a tool that consists of several different frames, each representing a different portion of a movement sequence. You can use the supplied animbrushes, or create your own. When you paint with one of these tools, the animated brush lays down a series of images within individual frames along the path that you specify with your brush stroke. When you play back the animation, you see realistic movement of the brush object.

So, let's get started. Just follow these steps:

- 1. Pull down the Selection menu and choose **Load as Selection.** This option allows you to load a file into an existing painting or animation, so that the image in the file automatically becomes your currently selected object—an easy way to keep the new object from merging with the underlying painting until you're ready to release it.
- 2. Select the Animation folder by double-clicking on it.
- 3. Select the Bird file and open it. Bird is an animbrush that Electronic Arts has prepared for you. To see the eight frames of animation it contains, press Command-8 repeatedly.
- 4. Select **Set # of Frames...** from the Anim menu and set the number of frames to 20. Click OK. You'll always need to choose a starting number of frames when creating an animation. You can always insert new empty frames later, or delete unused ones. Fig. 19.2 shows the Anim menu.
- 5. Select **Anim Move...** from the Anim menu.
- 6. Place the bird at a beginning position. (Use the mouse to select and move it.)

- 7. Choose the End Key and place the bird at an ending position.
- 8. Click OK and wait for the animation sequence to be drawn.
- 9. In the Control Panel at the bottom of the screen, set the speed to 20 frames/sec by moving the scroll bar.
- 10. Press the Continuous Play button (second from right in the Control Panel).
- 11. The bird will fly across the screen, looping until you press the space bar or click the mouse.
- 12. To clear your animation, select the **ClearAll Frames** option from the Anim menu.

Figure 19.2



Studio/1's Anim menu, which allows you to specify parameters like the number of frames to be used for a sequence.

Quick Start 2: Creating a Spinning Title

Spinning titles are an attention-getting device that you can use in your business presentations. In this exercise, you'll learn a quick way to produce one. This Quick Start will also be your introduction to the use of Studio/1 templates. Just follow these steps:

- 1. Press Command-1 to set the frame to #1 in the Control Panel.
- 2. Double-click on the text tool and choose a large font (larger than 18-point). Click OK.
- 3. Click on the screen to create a text box, and type a word. Click outside the text box.
- 4. Select **Anim Template...** from the Anim menu. Here you'll see a sample of some of the templates included with Studio/l.
- 5. Click on various templates to see their effect in the picture preview box.
- 6. Select any template and choose Draw.
- 7. When the drawing is complete, choose Continuous Play from the Control Panel to watch your animation.
- 8. To stop the animation, press the space bar or click the mouse.
- 9. To clear your animation, select "ClearAll Frame" from the Anim menu.

Quick Start 3: Creating a Bouncing Ball

Now you're ready to animate an object that you create yourself. We'll start with a simple ball that you can make bounce around the screen. Just follow these steps:

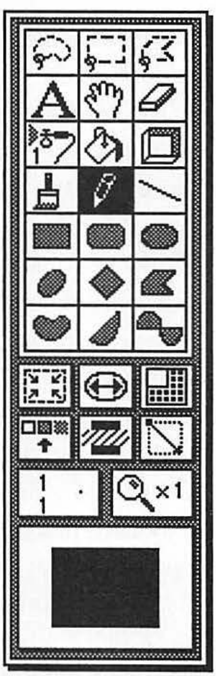
- 1. Press Commandl to set the frame to #1 in the Control Panel.
- 2. Select a gradient from the pattern palette and create a sphere with the circle tool. (After drawing your circle, you will be given a gradient fill line. Click to specify the fill direction.)

- 3. Turn the ball into a brush by choosing Last Object Brush fromthe Brush menu.
- 4. Hold down the tilde (~) key and paint with the ball. (New balls will appear as you continue AnimPainting beyond 20 frames.)
- 5. Choose Continuous Play from the Control Panel to watch your animation.

Other Features

Studio/1 provides a complete set of tools for painting, as shown in Figure 19.3. The first six include the selection tools, text, grabber hand, and eraser, all of which should be familiar to you from MacPaint-type programs that you have used in the past. The next three consist of an adjustable airbrush, a paintbucket, and a square that you can use to

Figure 19.3



Studio/1's painting tools.

isolate and "pick up" existing patterns from your painting. You should find the brush, pencil, and straight-line tools easy to use. You can use any of the 32 different brush shapes provided by Studio/1. Or, if you want, you can select any portion of the painting and use it as a brush. The nine polygon and curve tools let you draw selected shapes, either filled or unfilled.

In the middle of the tool palette are six modifiers, which determine how some of the other tools operate. For example, you can specify that selection tools shrink to fit an object, or expand to include all of a pattern within the surrounding background.

That's all you really need to get started with Studio/1—except for one thing: To exit the program, select "Quit" from the File menu.

Summary

This chapter introduced you to the concept of animation, and explained some of the challenges that arise in trying to get the Macintosh to display movement.

The chapter also described and explained how to use the Studio/1 demo program on the diskette included with this book. As a sophisticated monochrome painting program, Studio/1 is a good value. Some users may buy it just for its ability to handle resolutions up to 300 dpi, and for its PostScript-compatible text layer.

Apple Scanner users can control their scanners directly from within Studio/1. Others can scan with a Desk Accessory such as Scan/Do, or, if they use MultiFinder or System 7, with a scanner program loaded into another layer. Users can import these images through the Clipboard, or directly from one of the supported file formats.

Studio/1 also lets you animate the images you draw or scan. You don't have to tediously redo each in-between image, either. You can specify a start point and an end point for an object, along with the number of frames the motion should cover. Studio/1 generates all the images in between these points. Studio/1 also has special effects for transitions, fades, distortions, and sounds.

20

Introduction to QuickGIF and PCXtc

QuickGIF and PCXtc are utilities that let scanner users view two of the most popular image file formats: Graphics Interchange Format (GIF) and PCX. More advanced versions of these programs are available directly from their authors for a nominal fee. This chapter will get you started in using these programs, and will provide some hints on how they can help you with your scanned images.

What's GIF?

GIF is a very popular image file format, developed by CompuServe Information Service (CIS or, as addicts are prone to spell it, CI$), as a way for Macintoshes, IBM PCs, and other computers to exchange images. GIF supports monochrome, gray-scale, and color images.

CompuServe initially created GIF to bypass the need to duplicate each image in both a Mac and an IBM PC version. Even with the massive magnetic storage capabilities of the CIS mainframes, it would have been prohibitively expensive to store two copies of every image. As it stands, there is a mammoth collection of GIF images on CompuServe.

Because the format is well defined (unlike TIFF, which uses tags that more or less allow each file to define itself in new and potentially problematic ways), all you need is a viewer for each platform that understands the GIF file format.

GIF is quite sophisticated. It allows both interlaced and non-interlaced images, and has the ability to include single and multiple images in one file.

Because GIF files have already been compressed with LZW algorithms, there is no need to archive them further before uploading them to a bulletin board system (BBS) or to CIS. Conversely, a program such as UNSTUFFIT isn't necessary to extract GIF files. All you need is a viewer, such as QuickGIF. Because the program can load both PICT and GIF files and save in either format, you get the added ability to convert your scanned images to GIF, or to edit GIF files that you acquire.

What Is QuickGIF?

QuickGIF is a shareware application that can decode and display GIF graphics very quickly on Macintosh computers equipped with Color QuickDraw. The program can open and view GIF, PICT, and MacPaint files, and can play sound files produced by Macintosh sound digitizers. It can save image files in GIF or PICT format. Alternatively, you can copy the files that you are viewing to the Clipboard.

QuickGIF enables applying some simple transformations to your images, such as adding spheric, conic, mirror, oil paint, and dithering effects to black-and-white images. A photo mode lets you use the entire screen to display graphics.

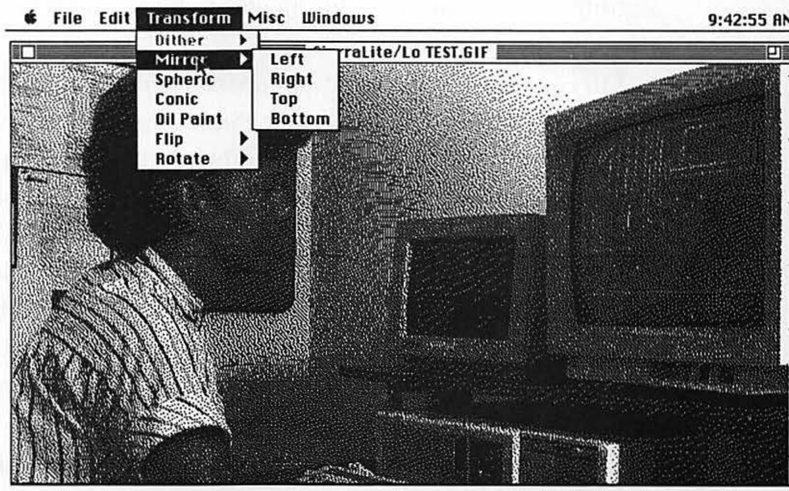
Using QuickGIF

Double-click on the application to launch it. The Open... menu allows you to choose multiple graphics files within a folder by shift-clicking on the names of the files you'd like to work with. When you click the Open button, all the selected files are opened in the order found.

You may view the files as you like. The Save As... menu lets you save the contents of the frontmost window as GIF files or in PICT format. QuickGIF is shown in Figure 20.1.

The application supports both interlaced and noninterlaced GIF images. The interlaced version includes every other line of the entire

Figure 20.1



QuickGIF

image, followed by the alternate lines. Some telecommunications software utilities enable displaying GIF files as they are downloaded from CompuServe or a BBS. With an interlaced file, you see a rough version of the entire image more quickly; a full thumbnail is visible when only half the image has been downloaded. If you're posting your scanned files on a BBS or CompuServe, you might want to use the interlaced format as a convenience for those who'll be downloading your images.

QuickGIF's transformations work with all graphics up to 8 bits per pixel (256 colors or gray tones). The program loads transformed images into new images without affecting the original image. You can experiment with several different transformations, leaving the original file untouched.

Available transformations include the following:

Dither. This conversion uses one of three different error dispersion algorithms to transform color images to black and white. (There is no effect on black and white images.)

Mirror. This transformation doesn't produce a reversed image of the entire graphic, as you might think. Instead, it produces a true mirrorlike effect, in which the left, right, top, or bottom part of the image can be projected onto the other half, just as if you had placed a mirror in the middle of the image.

Spheric. This transformation enlarges the pixels at the center of the graphics, generating a fisheye-type image that you will find interesting. Spherization works only on Macs equipped with a floating-point processor unit (FPPU), as the math involved in the transformation is quite complex.

Conic. The opposite of spherization, this transformation produces a conelike effect. An FPPU-equipped Mac is also necessary to produce conical effects.

Oil paint. This transformation produces the brush strokes of an oil painting on your image. A slow transformation, but quite interesting.

Flip and Rotate. These transformations let you and flip and rotate an image horizontally and vertically, producing a mirror image.

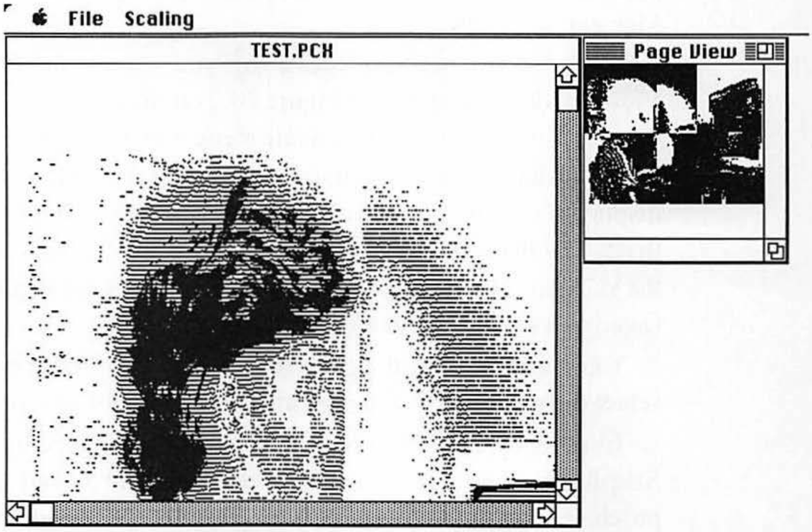
You'll find more information on QuickGIF, along with instructions for registering your copy, in the documentation included with the diskette accompanying this book. Registration information also appears in Appendix D.

For $30, you can receive a QuickGIF Plus diskette. This non-shareware program has the ability to create a slide show of graphics and files within a folder.

About PCXtc

PCXtc is a simple program that lets Mac users view monochrome PCX files produced by IBM PC programs, such as scanned line art or dithered gray-scale images. Publisher's Paintbrush and PC Paintbrush

Figure 20.2



PCXtc

IV Plus, both offered by ZSoft, can save scanned images in the PCX format. Likewise, other image editing programs can save in this format, providing a means of getting images to your Mac when you have no other recourse. The program is shown in Figure 20.2.

The challenge, of course, is that PCX images can be larger than the standard MacPaint display area (roughly a screenful, depending on your Mac display adapter). PCXtc provides two scaling options. **Squeeze to Fit** takes the PCX file, regardless of size, and scales it to fit the MacPaint standard. Some distortion will result, but may not be objectionable. **Normal scaling** shows the entire PCX image, which can be scrolled on the screen for viewing.

The program is simple to use. You launch it by double-clicking on its icon. PCXtc can open MacPaint files or PCX files. When you open a PCX file, the image will be converted in about 15 to 40 seconds, depending on the size of the file, its complexity, and the speed of your Mac and hard disk.

When the file has been converted, you see one main and one Page View window, as shown in Figure 20.2 on the previous page. If you're viewing a normal PCX file, a Scale menu is enabled. You can pull down this menu and select Normal to reveal a third, Full View window, which displays the entire file in miniature. A reversed rectangle shows the area that a MacPaint file could encompass. Some PCX files are larger than the standard MacPaint dimensions, while others are smaller and can be saved in a single MacPaint file.

You can click and drag the rectangle in the Full View window to select different parts of the picture for viewing or saving.

To save a PCX file in MacPaint format, you need a utility such as SnapJot to capture the area of the screen being displayed. Or, you should purchase the upgrade of PCXtc (version 1.6) from the program's author. Appendix D provides instructions on how to do this.

Summary

This chapter offered an introduction to the two file viewing and conversion utilities provided with this book, QuickGIF and PCXtc.

QuickGIF can load both CompuServe GIF files and PICT files, and save in either format. The program is a convenient utility for looking at image in either format, or for converting your scanned images to and from GIF and PICT.

PCXtc is a clever utility that gives Mac owners access to monochrome PCX files created by some IBM PC programs, such as Publisher's Paintbrush and PC Paintbrush IV Plus. A more advanced version of the program, available from the author, lets you save and print these files.

21

Gray Scale Editing with GrayView

GrayView is a gray scale editor for the Mac II with some interesting features. It can handle PICT, TIFF, MacPaint, and ThunderScan files, and can convert the latter to true 32-tone gray scales. If you don't already have a good gray scale editor, GrayView provides an excellent way to get your feet wet at low cost. Even if you do have a program like Digital Darkroom or ImageStudio, GrayView still can do some things the others can't. And the price is right.

GrayView isn't shareware; it's more like *orphanware.* At this time, there is no registration fee required, and the author, David Fry, has no plans to support or further enhance the program. You're free to use it for your personal work, but GrayView is copyrighted and can't be incorporated into any commercial project unless you get permission from as the author, as I did.

As you might guess, GrayView Version 1.9.8a4 is a commercial software project that never quite made it to market. As such, it's a true diamond in the rough: not fully polished, but admirable in its natural state. All the pieces haven't been completely connected, and not every feature works, but I think you'll find GrayView to be a valuable addition to your software library.

If nothing else, GrayView demonstrates that being ahead of your time isn't always a good thing. The original Mac II was introduced in March of 1987. Less than six months later, Fry, then a mathematics graduate student in Massachusetts, had completed a program which was able to display ThunderScan SCAN files in true gray scales on the Mac II.

That early version of GrayView (Version 1.16) could display and save PICT files, and could save images as StartupScreens. But, that was about all it could do.

In September of 1987, Fry, who had no experience or training in graphics programming, undertook the ambitious task of adding MacPaint-like features to his program. Gray scale imaging on the Mac was so new at the time that no one really knew what people would want from a gray scale editor. Nevertheless, Fry put several intense months into the effort, fully intending to sell the final version.

However, before he was finished, programs like ImageStudio were introduced, and programming was taking a lot of time from studies. Fry abandoned the project, and released the program to the public for "general use and playing."

According to Fry, "It has lots of missing features, though, things I never got around to completing before I quit programming. In particular, the dragging and pasting functions aren't quite what they should be."

The rest of this chapter will tell you how to use the key features of GrayView. I've adapted this material from David Fry's documentation, which is also included in unaltered form on the disk packaged with this book.

What is GrayView?

GrayView is an image editor especially designed to convert ThunderScan SCAN documents to gray scale images on a Mac II. This version allows the user to manipulate the picture in several fundamental ways, like a MacPaint for gray scales. GrayView can also be used to view and manipulate any PICT image, color or black-and-white, as well as some TIFF images.

GrayView should be used with a Mac II with a video board that can display 16 or 256 colors or grays. If your board can display more colors, you should set the display to 256 tones through the Macintosh Control Panel. In 16-color mode, images will display the banding phenomenon that is known as posterization. In addition, you won't be able to convert ThunderScan documents to 32 gray shades in this mode. For best results, you should use 256-color mode. GrayView's main screen is shown in Figure 21.1.

Figure 21.1

GrayView's main screen.

GrayView's Menus

This section will tell you about the key features of GrayView's Menus, working from the left side of the menu bar toward the right. I won't attempt to provide a comprehensive tutorial on using GrayView. This description is best suited for those who are already familiar with painting and image editing programs, and who just want to know how to make GrayView do its stuff.

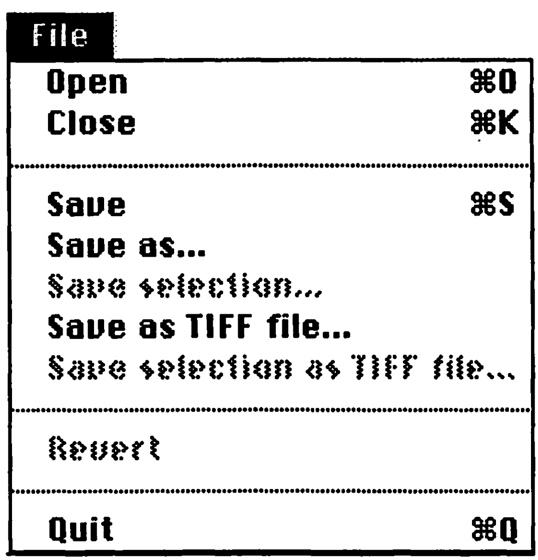
Apple Menu

The **About** box in the Apple menu contains information about the available memory that GrayView can use. Generally, available memory should be at least twice the size of the file on disk. If the file contains 200K, you'll need at least 400K of memory available to open it. There are additional memory demands, however, because of the program's overhead. GrayView will inform the user if memory runs low.

File Menu

The File menu shares many of the items found in every Macintosh program. However, you'll need to know some of the special features available to you. The File Menu is shown in Figure 21.2

Figure 21.2



GrayView File Menu.

Open

The "Open" item allows you to open five types of files: a document created with GrayView, PICT files, MacPaint files, ThunderScan SCAN files, or TIFF files. MacPaint files are opened using the current depth of the graphics device (16 or 256 tones). So, while MacPaint files start out as black-and-white, you can color them any way you like.

If you choose to open a SCAN or TIFF file, a window will be displayed and the file will be converted to gray scales immediately. You can watch the process, as the conversion is quick but not instantaneous. If you like, you can do something else, such as move and resize windows, or use a desk accessory.

However, you can't use any of the paint tools while the conversion is underway. If you're running under MultiFinder, you can place GrayView in the background and the conversion will continue. The conversion process can be stopped at any time by hitting command-(period), or the ESC key.

GrayView can look at any PICT resource. If you hold down the option key while choosing Open, GrayView will display *all* the files on disk and you can choose one in which GrayView will search for a PICT resource with ID = 0.

If it finds one, it will display it; if not, it will tell you that. PICTs with ID=0 are special on a Mac II because such a file can be made a StartUpScreen at boot time by naming it "StartUpScreen" and placing it in the System Folder.

Close

The "Close" item will close any active window or desk accessory, asking you to save changes if necessary.

Save, Save As...

The "Save" and "Save As..." items allow the user to specify the file to be saved as a standard GrayView file, (the default type), or as a PICT file.

Save Selection

The "Save Selection" item allows the user to save a portion of the file currently selected with the marquee or lasso.

Save as TIFF file, Save selection as TIFF file

The "Save as TIFF file" and "Save selection as TIFF file" items allow the user to save the current window or a portion of it as a TIFF file for exporting to applications which read this format. For instance, a picture can be saved as a TIFF file and printed on the LaserWriter using PageMaker.

TIFF files are saved with the same depth as the window; if the window was opened with 256 colors active, the TIFF file will contain 256 gray shades. If the window was opened with only 16 colors, the TIFF file will only have 16 gray shades.

Colors in the current window are converted to grays using the same formula the Mac uses to display color images on black-and-white monitors. So if you have some color in the picture, you can see what the TIFF output will look like by setting the Control Panel Monitor tool to "Grays" temporarily.

Revert

"Revert" will open the last saved copy (if one exists) of the current window.

Quit

"Quit" allows the user to leave the program.

Edit Menu

Not all the items in this menu have been implemented in GrayView. However, you can generally carry out the functions you really need to. The Edit Menu is shown in Figure 21.3.

Undo

"Undo" is fully supported by GrayView. You can use it to undo and redo any paint action or image manipulation (described below).

NOTE: Because of MacPaint we've all become used to the upper left key meaning Undo, so the Esc key (for the regular Apple keyboard) and the "' key (for the Extended Apple keyboard) are synonymous with the Undo menu item, as is its command key equivalent command-Z.

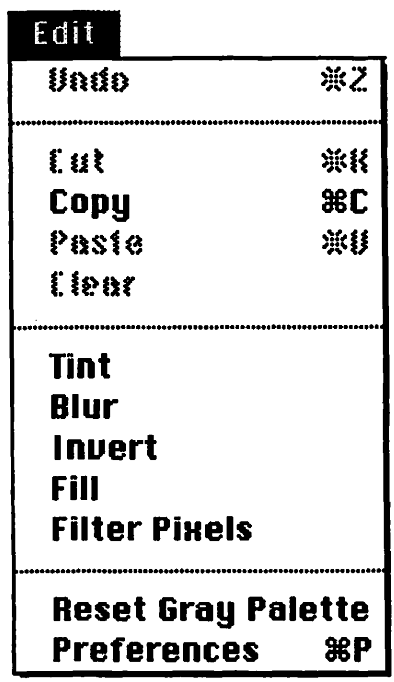
Cut

"Cut" is not supported by GrayView.

Copy

"Copy" will copy a selected portion of the file, using the marquee or lasso, to the clipboard.

Figure 21.3



GrayView's Edit Menu.

Paste

"Paste" will take any picture data currently on the clipboard and paste it into a new window.

Clear

"Clear" is not supported by GrayView.

Other choices

The rest of the Edit menu consists of commands for manipulating the appearance of the current image. They will all work with color pictures but the results are not always as you'd expect; they are designed for gray scale images. Play around, though; you can always Undo later.

Tint

The "Tint" item is used to tint any selected portion of the image with the currently selected color or gray on the gray palette. This process substitutes the color in place of the image wherever the image is darker than the color.

Making the color darker (by double clicking on it and using the Color Picker) will cause less of the image to be tinted. This is easily seen by experimentation. If no portion is selected, Tint is applied to the whole image.

Blur

The "Blur" item will soften the focus on any selected area, making it look like it's under water. This is useful for blurring sharp borders between objects. This is a rather slow process and may seem to take a little while if the selected area is large. The spinning hands on the watch cursor will let you know that something really is happening. If no area is selected, Blur is applied to the whole image.

Invert

The "Invert" item is used to invert any selected portion of the image. If no portion is selected, the inversion is applied to the whole image.

Fill

The "Fill" item is used to fill any selected portion of the image with the currently selected color or gray on the gray palette (see below). If no portion is selected it is applied to the whole image.

Filter Pixels

Sometimes a scanned image has imperfections, especially random noise in darker areas. This was particularly true of early ThunderScan pictures. The "Filter Pixels" item will attempt to remove such dirt from any selected portion of the image. This is generally desirable, but sometimes it will filter the wrong things, perhaps removing the sparkle from someone's eye. Fear not, however, for it is undoable, like everything in GrayView. If no area is selected it is applied to the whole image.

The Blur and Filter Pixel operations are not instantaneous, but you can see the progress in the window. If you decide you don't want to wait for the process to finish, click the mouse and it will stop where it is.

Reset Gray Palette

The "Reset Gray Palette" item will reset the gray scale palette on the left portion of the screen to its default setting.

Preferences

The "Preferences" item brings up a dialog box allowing you to make certain choices. The "default FatBits level" and "color matching sensitivity" will be discussed below.

Certain pictures need special colors or grays available to be displayed properly, but they might not necessarily be available to GrayView. For instance, GrayView uses 64 levels of gray to display digitized images, but if you try to open a color digitized picture, those 64 grays will get in the way of more useful colors, such as light skin tones.

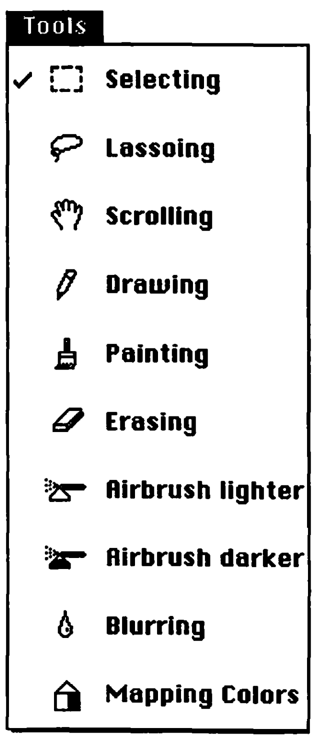
If you turn on the "Use documents native colors" option in the Preferences box, GrayView will look inside any GrayView or MacDraw PICT document, or any picture pasted from the clipboard, and search for the colors that will make that picture look best. These will be made available before putting up the window with that picture. Thus specialized pictures can be shown much more faithfully.

The Preferences box also has a "Use 'best colors' in 16 colors" option which is not at all operational yet. It will allow users with limited colors palettes to view digitized pictures better, even those that require more than 16 grays.

Tools Menu

This merely makes the selected tool the current tool. It is useful if the tools palette is currently covered by a window, and the menu names remind you what each tool is for. The Tools Palette is shown in Figure 21.4.

Figure 21.4



GrayView Tools Menu

Windows Menu

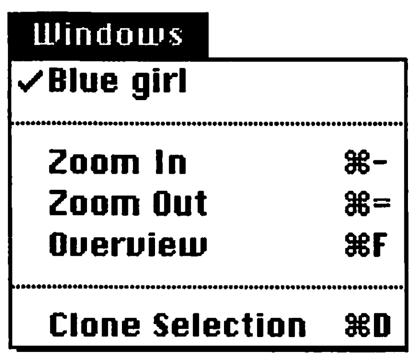
The Windows Menu is shown in Figure 21.5.

The first portion of this menu contains the names of the open windows. Selecting a non-active window will make it the active one.

The second part of the menu is for Zooming in and out of a document. This allows you to look at images at magnifications of 1X,2X,4X, and 8X. Selecting "Zoom In" moves you to the next magnification level, whereas "Zoom Out" decreases one level.

At any magnification other than 1X, the current window is split in half, with the magnified portion to the right and the normal portion to the left. Clicking on the normal portion will cause the area surrounding the pointer to be displayed at the right. You can click on the normal

Figure 21.5



Grayview Windows Menu

portion and move the mouse around (while holding down the button) to scroll in any direction.

You can use the Preferences box to specify a "default FatBits level," either 2X, 4X, or 8X. Then if you double click on the Pencil tool (see below) you will go to this FatBits level if you're not already there. If you are at the FatBits level, you'll be returned to normal 1X level. Also, command-clicking with the Pencil or the Paintbrush on an open window, will cause that area (where you clicked) to be displayed in FatBits if it is not already, or returned to normal 1X mode if you're already at FatBits.

Any of the palette tools described below can be used on the magnified portion, as well as any of the effects from the Edit menu.

Overview

"Overview" displays a version of the active window so that the image fits inside a alert box on the screen. A frame is also displayed to indicate what portion is currently visible. You can you use the mouse to position the frame to view another portion of the document. This is useful for looking at the whole picture if it is larger than the screen.

You can also get the Overview by double-clicking on the Hand tool on the tools palette.

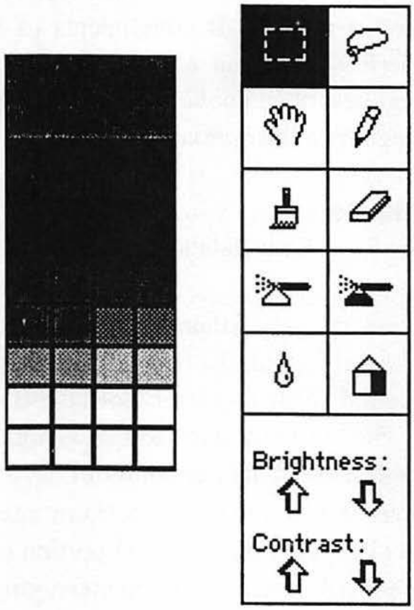
Clone Document

"Clone Document" opens a window with another copy of the active window. The user is given the chance to specify scaling factors for the new version, so you can make it larger or smaller if you wish. You can clone a specific selection in the active window by selecting it with the marquee tool and choosing "Clone Selection."

The Gray and Tool Palettes

The palette on the left side of the screen initially shows the 32 grays used for ThunderScan conversion. If you are not in 8-bit mode, all available grays are shown, with the remaining colors shown as white. The Gray and Tool Palette is shown in Figure 21.6

Figure 21.6



GrayView's Gray and Tool Palettes.

At the top of the palette is the currently selected color, used for drawing, paint, filling, and tinting. Clicking on any of the other colors makes it the current color.

Double clicking on any color (including the currently selected color) will cause a small window to pop up next to the mouse with all possible colors available. The color (or gray) you double-clicked on will be highlighted on the window.

Use the mouse to select a new color and let go of the button. This new color will replace the one you double-clicked on. If you decide instead to retain the original color selection, click anywhere outside the window.

Sometimes, however, you may want to use the standard Apple Color Picker package to choose your color. For instance, if you want exact control of the R-G-B components of the color, the Color Picker is wonderful. You can use the Color Picker rather than the window method described above by double clicking on the palette color while holding down the option key.

Reset Gray Palette

The Reset Gray Palette menu item will reset the colors to their default grays.

Below the gray palette is the tool palette, consisting of the marquee, the lasso, the hand, the pencil, the eraser, the paint brush, the airbrush-lighter and airbrush-darker tools, the Blur tool, and the Color Mapping tool. The currently selected tool is highlighted. Clicking on any other tool will make it the current tool.

Note: if you attempt to perform any of the actions under the Edit menu (Tint, Fill, etc.) and NO portion of the document is selected, the program will interpret that to mean you want the whole document, so the marquee will be made the current tool and the entire picture will be selected, including any portion not shown in the window.

Also, if the cursor is over the active window and you hold down the option key, the cursor will change to a hand cursor and you can use it to drag the document around, instead of using the scroll bars.

Selection Tools

The marquee and lasso work like their MacPaint equivalents. If you are selecting near the edge of a window, the window will scroll to reveal more of the image (if there is any available, that is). Holding down the option key while scrolling an image will activate the fast scrolling mode for selecting large areas.

When a region has been selected with the marquee or the lasso and you move the pointer over the region, the pointer changes to the normal arrow. Now you can click on this region and drag it anywhere, just like MacPaint. See the end of this file for warnings about dragging. If you hold down the shift key a copy of the selected region will be dragged, otherwise the selected region will be erased to white first.

Pencil Tool

The pencil is used to draw in the current color. This is nice for touching up scanned documents. The pencil has two other very important features: first, if you hold down the shift key while clicking in a window with the pencil tool, the color underneath the pencil will be picked up and made the current color. This color is also displayed on the tool palette in place of the Pencil tool while the shift key is down. If you continue to hold the mouse down it will draw in this new color. It's great for easily removing blemishes: suppose a person in your image has a mole on his nose that you want to remove. The skin tones surrounding the mole will be lighter than the mole itself. Therefore, hold down the shift key and click on the skin near the mole. This lighter color will be made the current color. Now draw over the mole, and presto!, no mole. This trick is also good for using the exact color or gray located elsewhere in a document (or in another document, for that matter).

Secondly, holding down the Control key while clicking in the document indicates that you want to Tint the document with the pencil rather than draw. Moving the mouse around then will have the same effect as the Tint menu item.

Hold down the shift key and the Control key and the pencil tool will Tint with the color right underneath where you initially clicked on the window.

If you double-click on the pencil tool you will be taken to the default FatBits mode if you're not already there, and to normal mode if you are already in FatBits.

Paint Tool

The paint tool is identical to the pencil, just using a bigger drawing area instead of a single point. It also uses the shift trick to pick up colors and the Control key trick to Tint.

If the pencil or paintbrush is selected, command-clicking on the active window will take you to FatBits, with that area centered on the FatBits display.

Finally, double-clicking on the paint tool allows you to select a new paintbrush shape.

Eraser Tool

The eraser tool is similar to the paint tool, but it erases the picture to white, like MacPaint.

Airbrush Tools

The airbrush-lighter tool will lighten pixels underneath it while the mouse is held down. The longer the mouse is down the lighter things become, so be sure to move the mouse about so as to not "burn" the image. The airbrush-darker tool darkens pixels, working the same way as the airbrush-lighter tool. The effects of each of these tools can be very subtle but it is really doing something. Just try leaving the mouse in one place for a few seconds to see this.

You can change the size of the area affected by the airbrush tools by double clicking on the tool in the palette.

Blur Tool

The "Blur" tool is used to blur areas with the mouse. Moving it around (with the button down, of course) constantly applies the same softening algorithm used in the Blur menu item. This way you can selectively soften the focus on areas, make borders between objects less distinct, or bleed colors or grays together, as if water was applied to a watercolor painting.

Color Mapping Tool

The "Color Mapping" tool can be used to replace all instances of any color (or gray, which is what it is primarily used for) in the selected region with another. To do this, first use the marquee or the lasso to select some area (if you don't, the tool assumes you want to remap the whole image). Also, change the current color (at the top of the palette) to the color you want to map to (i.e., the color you'll be left with after remapping). Then choose the Color Mapping tool, and click on a pixel in the document which has the color you wish to be remapped. The color underneath the Mapping tool is displayed in the tool's place on the palette to make it easier to choose which pixel to click on. Once you click, all instances of the color under the tool which occur in the selected area will be replaced by the currently selected color.

The Preferences dialog box allows you to specify the "color mapping sensitivity." This is a percentage between 0% and 100%. The idea is to determine how picky the mapping tool is going to be. For 100%, the tool is completely sensitive and it will only change pixels which have the *exact* same color as the pixel you clicked on. But you can change that; if you set the sensitivity to 95%, say, it will change pixels whose color is relatively close to that of the pixel you clicked on. Exactly how close 95% means the color has to be is hard to say, but the effects are obvious if you experiment with the color mapping tool and sensitivity. NOTE: a 0% sensitivity means that all colors in picture will be changed and color mapping tool has the same effect as Fill.

A note is necessary about the pencil, the paint brush, airbrush, and the Blur tools: these tools will generally be used only on small regions between mouse clicks. If you use the paint tool, for instance, to paint a large sweeping area and memory is low, the program may decide that the action would not be undoable. In this case, it would beep once and the action would be undone right then. The correction for this is to paint the same area as a series of smaller areas, lifting the mouse button periodically. Then each smaller area would be undoable until the mouse was clicked again. Please do not worry about this unless you are running low on memory. Generally it is not a problem.

Brightness and Contrast Controls:

Below the tool palette you can find four arrows, two for each of the "Brightness" and "Contrast" controls. You can use these by selecting a region of the current window using the marquee or lasso tools, and clicking on the arrows (If you don't select a region the controls presume you want the effect to take place over the whole document). Clicking on the up brightness arrow will make the region lighter, clicking on the down brightness arrow makes it darker. The contrast control works similarly: click on the up arrow to increase the contrast and the down arrow to decrease the contrast.

If you continue to hold the mouse button down on this control you can continue to change the picture. By holding the mouse down on the up brightness arrow, for example, you will see the selection get lighter and lighter until it is eventually white.

Each of these controls are designed to work with B/W pictures. They may or may not work with color, depending on the nature of the color picture.

There's a special trick for the Brightness controls. By holding down the option key while clicking, you can invoke the "Wrap Around" mode. For instance, if you option click on the up brightness arrow, the picture will get lighter and lighter. But any pixels that become totally white will "wrap around" to black. If you option click on the down brightness arrow blacks will wrap around to white. This is not very natural but it does make for some nice effects.

Final Considerations

Here are some limitations of GrayView in its current (and so far final, but unfinished version).

- Doesn't read ThunderScan TIFF files.
- Only reads 4 and 8 bit TIFF files.
- Only reads TIFF files while 256 colors are available.
- Save Selection as TIFF file will often mess up the screen by drawing the marquee askew.
- Needs to learn how to request temporary memory from MultiFinder.
- When memory is short a window will sometimes open blank.

- Doesn't save 4 bit TIFF files properly when using the 16 color setting in the Control Panel.
- Dragging: Dragging functions are still in the early stages. First, dragging doesn't work at all in 2X, 4X, or 8X modes. Second, if you drag a region and then let go of the mouse it will remain selected, which is correct. But if you try to drag it again it will erase what was underneath it. Also, the Paste function must be modified so that the picture on the Clipboard is pasted into the active window and not a new window (which will be preserved as an option). Finally, when a dragged region is erased it is replaced by white. The user will be able to choose the color that should replace the selected region (it will be called the background color).

Summary

GrayView is a useful gray scale editing program that you'll have fun playing with. It can work with PICT, TIFF, MacPaint, and ThunderScan SCAN files, and allows you to modify brightness and contrast, manipulate individual pixels, and convert from one file format to the other.

If nothing else, GrayView will let you learn about gray scale editors and do some experimenting. It will also provide you with an inside look at a product under construction, so you can appreciate what software authors go through in bringing their brainchildren to market!

Appendix A

Glossary

Appendices in general are often under-rated. Just as the human appendix is an extraneous, digestive system dead-end, an appendix in a book is sometimes seen as a place to tack on superfluous information that wasn't important enough to be included in the main body of the book.

This appendix, however, and those that follow are designed to be valuable additions to your scanner toolkit. Here, you'll find definitions of a high percentage of the specialized terms and jargon you'll encounter when using scanners, desktop publishing, and pre-press systems. You can use this word list to refresh your memory or find explanations of topics that aren't totally clear to you.

I did try to define many of these terms in the chapters in which they first appear. However, it's not practical to explain every term without bogging down the text with continual digressions. Beginners who find a few chapters a little confusing can use this glossary to get up to speed quickly. You'll also find some words that weren't used in the text at all but which are commonly used in the industry.

Achromatic color: A color with no saturation.

Additive colors: The primary colors of light—red, green, and blue—which, when combined, produce white light.

Airbrush: An atomizer used for spraying paint. Image editing software usually has an airbrush-like tool which can apply a fine spray of a given tone to a specified area. Fully controllable airbrushes allow you to adjust the size of the airbrush spray, its density or concentration, and the speed at which the spray flows.

Anamorphic: An image that has been enlarged or reduced more in one direction than another. The image looks "squashed" or "stretched" in a given dimension.

Antialiasing: A process that can be used to remove jaggies or stair-stepping in an image. Antialiasing smooths out diagonal lines by placing dots of an in-between tone in appropriate places.

Applications program: Software, such as a word processing program, spreadsheet, or database manager, that performs useful work not directly related to the maintenance or operation of the computer. Ventura Publisher and PageMaker are applications programs.

Applications program interface: A common interface that allows software engineers to write programs that will operate with a broad range of computer configurations.

Archive: To store files that are no longer active. Programs like STUFFIT combine and compress files into an archive file for more compact, easier storage.

Aspect ratio: The relative proportion of the length and width of an image. For example, if you scan an original that measures 4 x 6 inches, it will have an aspect ratio of 4:6 or 2:3. To maintain the same proportions, you must place it in your desktop publishing document with dimensions that conform to the same ratio. That is, it could be sized at 2 x 3 inches, 1.5 x 2.25 inches, etc. CRT screens and printers also have aspect ratios.

Asynchronous: A communications method under which exact timing of the signals is not critical: the next set of information is sent whenever a confirmation signal is received. This is the opposite of synchronous communications, which send out data within an exact block of time. Macintoshes most commonly use asynchronous communications—with modems, for example—to exchange data over distances of more than a few feet. Some scanners communicate with your computer in this way.

Attribute: Characteristics of a page or character, such as underlining, boldface, or font, which can be captured by an OCR program.

Automatic document feeder (ADF): A device attached to a scanner that automatically feeds one page at a time, allowing the scanning of multiple pages.

Autotrace: A feature found in many object-oriented image editing programs that allows you to trace a bit map image and convert it to an outline or vector format.

Background: The ability to run unattended while another program is executing. On the Mac, background printing and telecommunications are most often used.

Back up: To make a copy of computer data as a safeguard against accidental loss. The copy that is made is called the backup.

Baseline: An imaginary line on which all the characters in a line rest.

Baud: A data transmission rate of 1 bit per second, used to measure asynchronous communications speed.

BBS: Bulletin Board System, a computer system that has been set up to function as a clearing house for the exchange of information among other computer users via modem. Service bureaus often set up a BBS to allow transmitting PostScript files for output directly to the bureau.

Bezier curve: A cubic polynomial in mathematical terms or, simply, a way of representing a curve that allows great flexibility in manipulating the curve. Bezier curves are adjusted using endpoints and anchor points.

Bilevel: In scanning, a binary scan that stores only the information that tells whether a given pixel should be represented as black or white.

Binary: Base-two arithmetic, which uses only 1s and 0s to represent numbers. 0001 represents 1 decimal, 0010 represents 2 decimal, 0011 represents 3 decimal, and so on. In scanning, a black-and-white image.

BIOS: Basic Input/Output System. A piece of computer code, provided on read-only memory (ROM) chips and used to govern basic system level functions.

Bit: A binary digit — either a 1 or a 0. Scanners typically use multiple bits to represent information about each pixel of an image. A 1-bit scan can store only black or white information about a pixel. A 2-bit scan can include four different gray levels or values—00, 01, 10, or 11. Other values include:

4 bits	**16 gray levels**
5 bits	**32 gray levels**
6 bits	**64 gray levels**
7 bits	**128 gray levels**
8 bits	**256 gray levels**

Bit map: A representation of an image in row and column format in which each individual pixel is represented by a number. A single bit or up to as many as 32 can be used with each increment representing a larger amount of gray or color information about the pixel.

Black: The color formed by the absence of reflected or transmitted light.

Black printer: The plate used for the black ink in the four-color printing process. It provides emphasis for neutral tones, detail in shadow areas of the image, and a deeper black than can be provided by combining cyan, magenta, and yellow alone.

Black printers can take two forms. A skeleton black adds black ink only to the darker areas of an image. A full-range black printer adds some black ink to every part of the image.

Bleed: An image that continues to the edge of the page. It is often accomplished by having the image extend past the edge and then trimming the page to the finished size.

Blend: To create a more realistic transition between image areas. Image editing software will often allow you to merge overlapping sections of images to blend the boundary between them.

Boot: To start up a computer.

Brightness: The balance of light and dark shades in an image. See also luminance.

Buffer: An area of computer memory set aside to store information meant for some sort of I/O, such as printing or writing to disk. This allows the device supplying the information to feed it into memory faster, if necessary, than the device meant to accept it can handle it.

A printer buffer, for example, allows an applications program to quickly dump a document for printing and then go on to something else. The buffer can then feed the information to the printer at a slower rate. In scanning, buffers are often used to store images awaiting processing.

Bug: An error in a program that results in some unintended action.

Burn: In photography, to expose part of a print for a longer period, making it darker than it would be with a straight exposure. In lithography, to expose a printing plate.

Bus: A hardware interface used to connect a computer to peripherals or other computers. You'll often see references to the SCSI bus or NuBus, which are both used by Macintosh computers.

Byte: Eight bits, which can represent any number from 0000000 to 11111111 binary (0 to 255 decimal).

Cache: A memory buffer used to store information read from disk to allow the operating system to access it more quickly. Cache programs use various schemes to make sure that the most frequently accessed sectors, as well as the most recently accessed sectors, remain in the buffer as long as possible.

CAD: Computer-Assisted Design. Also called Computer Aided Design and Computer Aided/Assisted Drafting/Design (CADD). A technique for creating engineering drawings and similar materials on a computer using line-oriented techniques.

Calibration: A process used to correct for the variation in output of a device like a printer or monitor when compared to the original image data you get from the scanner.

Camera ready: Art work that is printed in hardcopy form which can be photographed to produce negatives or plates for printing.

Cast: A tinge of color in an image, particularly an undesired color.

CCD: Charge-Coupled Device. A type of solid state sensor used in scanners and video capture devices. Compared to older imaging devices, including video tubes, CCDs are more sensitive and less prone to memory problems that can cause blurring of images.

CD-ROM: Compact Disk-Read Only Memory. An optical disk device that uses pits written on the disk by laser to convey bits of information. CD-ROMs are encoded with information during manufacture and cannot be written to by the user. They provide a means of distributing large databases on a compact medium.

CDEV: A Control Panel device, such as General, Keyboard, and Monitors, used to configure your Macintosh. Prior to System 7, CDEVs were all accessed from a single desk accessory with individual scrolling icons for each. Under System 7 you can access each Control Panel directly.

Character: An alphanumeric symbol, punctuation mark, or other symbol available from the PC keyboard.

Chooser: The desk accessory used to select from devices such as printers, and direct their communications through either the printer or modem port.

Chrome: Photographer-talk for a color transparency.

Chroma: Color or hue.

Chromalin: The DuPont trademark for a type of color proof used for representing how color halftones will appear on the printed page.

Chromatic color: A color with at least one hue available, with a visible level of color saturation.

Clip art: Artwork that is purchased or otherwise available for scanning or other uses in desktop publishing with few restrictions.

Clipboard: A memory buffer that can hold images and text so they can be freely interchanged within and between Macintosh applications.

Clipping: Compressing a range of values into a single value, as when a group of highlight tones are compressed to white or a set of shadow tones represented as black.

Clone: In image editing, to copy pixels from one part of an image to another.

Color: See Hue.

Color correction: Changing the color balance of an image to produce a desired effect, usually a more accurate representation of the colors in an image. It is used to compensate for the deficiencies of process color inks, inaccuracies in a color separation, or an undesired color balance in the original image. Color correction is done using one of several available color models, including RGB (red-green-blue) and LHS (luminance-hue-saturation).

Color key: A set of four acetate overlays, each with a halftone representing one of the colors of a color separation and tinted in that color. When combined, color keys can be used for proofing color separations.

Color separation: The process of reducing an image to its four separate color components—cyan, magenta, yellow, and black. These separations are combined using an individual plate for each color on a press. To create a color other than the three primaries, plus black, percentages of them are combined.

Color wheel: A circle representing the spectrum of visible colors.

Comp: A layout that combines type, graphics, and photographic material, also called a composite or comprehensive.

Compiler: A program that translates source code written in a higher level language into machine language object code.

Complementary color: Generally, the opposite hue of a color on a color wheel, which is called the direct complement. For example, green is the direct complement of magenta.

There are also two other types of complements: the split complement (a color 30° away from the direct complementary color) and the triadic (a color 120° in either direction from the selected color.)

Compression: Packing of a file or image in a more efficient form to improve storage efficiency. Compression and decompression take some time, so it takes longer to save and open compressed files.

Concatenate: To add together.

Contiguous: In reference to hard disks, contiguous sectors are those that are arranged consecutively on the disk. Your system software tries to allocate sectors to a file contiguously so that the disk drive can read as many sectors of a file as it can with a minimum of read/write head movement. However, as a hard disk fills, the unallocated sectors gradually become spread out and fragmented, forcing the operating system to choose more and more non-contiguous sectors. Fragmented files can be much slower to access.

Continuous tone: Images that contain tones from black to white with an infinite range of variations in between.

Contrast: The range between the lightest and darkest tones in an image. A high contrast image is one in which the shades fall at the extremes of the range between white and black. In a low contrast image, the tones are closer together.

Control character: A nonprinting character used to send information to a device, such as the control characters used to communicate special formatting commands to a printer.

Copy dot: Photographic reproduction of a halftone image, in which the halftone dots of a previously screened image are carefully copied as if they were line art. The same technique can be used in scanning to capture a halftoned image. If the original dot sizes are maintained, the quality of the finished image can be good.

Creator code: A four-letter code used by your system software to keep track of which application was used to generate a given document. This code allows the Mac to launch the right application when you double-click on a file. Since many applications can create several different types of files, a second code, called a type code, is used to differentiate among them.

Crop: To trim an image or page by adjusting the side or boundaries.

Crop mark: A mark placed on a page that is larger than the finished page to show where the page should be trimmed to final size.

Current directory: The default directory that DOS assumes you mean unless you explicitly type some other within a command.

Cursor: A symbol that indicates the point at which the next action the user takes – text entry, line drawing, deletion, etc. – will begin; the current screen display position.

CMYK: The abbreviation for cyan, magenta, yellow and black.

CYMK color model: A model that defines all possible colors in percentages of cyan, magenta, yellow, and black.

Daisy-chain: To connect peripheral devices in series, as with the SCSI bus. You cal also daisy-chain Apple Desktop Bus (ADB) devices, such as the keyboard and mouse.

Darken: A feature found in many image editing programs that allows gray values in selected areas to be changed, one value at a time, from the current value to a darker one. This is equivalent to the burning procedure used in conventional darkrooms.

Data compression: A method of reducing the size of files, such as image files, by representing the sets of binary numbers in the file with a shorter string that conveys the same information. Many image editing programs offer some sort of image compression as an optional mode when saving a file to disk.

DCA/RFT: A file format that is used as a near-universal interchange format among word processing programs. It is an acronym for Document Content Architecture/Revisable Form Text and was developed by IBM.

Decolumnization: Disassembling the text on a page so side by side columns are converted into a single column for interpretation by an OCR program.

Default: A preset option or value that is used unless you specify otherwise.

Descender: The portion of a lowercase letter that extends below the baseline. The letter p is an example of a character with a descender.

Diffusion: The random distribution of gray tones in an area of an image, often used to produce a mezzotint effect.

Digitize: To convert information, usually analog information, such as that found in continuous tone images (or music), to a numeric format that can be accepted by a computer.

Digitizer: A device that converts analog data to numeric format. Video cameras are a sort of digitizer. Another type of digitizer would be an input pad that enters coordinates of an image into a computer (usually found in CAD applications).

Dot etching: A technique in photographic halftoning in which the size of the halftone dots is changed to alter tone values.

Dithering: A method of simulating gray tones by grouping the dots shown on your CRT display or produced by your printer into large clusters of varying size. The eye merges these clusters and the surrounding white background into different tones of gray.

Dot: A unit used to represent a portion of an image. A dot can correspond to one of the pixels used to capture or show an image on the screen, or groups of pixels can be collected to produce larger printer dots of varying sizes to represent gray.

Dot gain: The tendency of a printing dot to grow from the original size when halftoned to its final printed size on paper. This effect is most pronounced on offset presses using poor quality papers, which allow ink to absorb and spread.

Dots per inch: The resolution of an image, expressed in the number of pixels or printer dots in an inch. Abbreviated dpi.

Download: To receive a file from another device. For example, soft fonts are downloaded from your computer to your printer.

Driver: A software interface used to allow an applications program to communicate with a piece of hardware, such as a scanner.

Drop cap: The first letter of a paragraph, set in a larger point size than the rest of the text. It may rise above the first line or extend below, in which case the drop cap is inset into the text block.

Dummy: A rough approximation of a publication, used to gauge layout.

Duotone: A printed image, usually a monochrome halftone, which uses two different colors of ink to produce a longer range of tones than would be possible with a single ink density and set of printer cells alone.

Dynamic RAM: Type of memory that must be electrically refreshed many times each second to avoid loss of the contents. Macintoshes use dynamic RAM to store programs, data, video information and the operating system.

EBCDIC: Extended Binary Coded Decimal Interchange Code. A code system like ASCII, used with IBM mainframes and some software, such as DisplayWrite. Supported by some OCR software.

Emulsion: The light-sensitive coating on a piece of film, paper, or printing plate.

Emulsion side: The side of a piece of film that contains the image, usually with a matte, nonglossy finish. This side is placed in contact with the emulsion side of another piece of film (when making a duplicate) or the printing plate. That way, the image is sharper than it would be if it were diffused by the base material of the film.

Scanner users need to understand this concept when producing images oriented properly (either right-reading or wrong-reading) for production.

Encapsulated PostScript: An outline-oriented image format that represents graphics and text in terms of descriptions of how to draw them. Desktop publishing programs like PageMaker, QuarkXpress, Ventura Publisher, and DesignStudio can import these files, while vector-oriented draw programs can often modify them.

Export: To transfer text or images from a document to another format. Some applications provide a Save As... option to save the entire file in the optional format, while others, such as GrayView, let you save a selected portion of the image or file in another file format.

File: A collection of information, usually data or a program, that has been given a name and allocated sectors by operating system.

File name: The name given a file, which can be quite long in the Macintosh environment, but is, in contrast limited under MS-DOS to just eight characters and a three-character extension.

File format: A set way in which a particular application stores information on a disk. This standardization makes it possible for different applications to load each others' files, since they know what to expect from a predictable file format. PICT, TIFF, and StartupScreen are all file formats found on the Mac.

Filter: In scanning, image filters are used to process an image—to blur, sharpen, or otherwise change it. Programs like Adobe Photoshop have advanced filters that will spherize, change perspective, and add patterns to selected portions of the image.

Finder: The part of the system software that takes care of opening, closing, renaming, moving, and erasing files and folders from your desktop. The Finder also formats (initializes) and ejects disks.

Fixed disk: Another name for a hard disk drive, so-called because such disks are not commonly removed from the computer while in use.

File-oriented backup: Any backup system that stores information in files, just as they are stored on the disk. Such systems allow easier access to and restoration of a particular file.

Flat: A low contrast image. Also, the assembled and registered negatives or positives used to expose a printing plate.

Font: A group of letters, numbers, and symbols in one size and typeface. Garamond and Helvetica are typefaces; 11-point Helvetica Bold is a font.

FPO: For Position Only. Artwork deemed not good enough for reproduction, used to help gauge how a page layout looks.

Format: To initialize or prepare a disk for use by writing certain information in magnetic form. Formatting divides the disk into tracks and sectors and sets up a directory structure, which is shown in the Macintosh world as folders and icons.

Four-color printing: Another term for process color, in which cyan, magenta, yellow, and black inks are used to reproduce all the hues of the spectrum.

Frame grabber: A device that captures a single field of a video scanner or camera.

Frequency: The number of lines per inch in a halftone screen.

Galley: A typeset copy of a publication used for proofreading and estimating length.

Gamma: A numerical way of representing the contrast of an image, shown as the slope of a line showing tones from white to black.

Gang scan: The process of scanning more than one picture at a time, used when images are of the same density and color balance range.

Gigabyte: A billion bytes of information; a thousand megabytes. Only ten 8.5 x 11-inch full color images scanned at 600 dpi would to fill up a gigabyte of disk space.

Gray balance: The proportion of ink in each of the three process colors (cyan, magenta, and yellow) which will combine to produce a neutral gray color.

Gray component removal: A process in which portions of an image which have all three process colors have an equivalent amount of replaced by black to produce purer, more vivid colors.

Gray scale: The spectrum of different gray values an image can have.

Gutter: The inner margin of a page which must be included to allow for binding.

Halftoning: A method for representing the gray tones of an image by varying the size of the dots used to show the image.

Handles: Small squares that appear in the corners (and often at the sides) of a square used to define an area to be scanned or an object in an image editing program. The user can grab the handles with the mouse cursor and resize the area or object.

Hardware: The physical components of a computer system, including the CRT, keyboard, microprocessor, memory, and peripherals.

Hexadecimal: The base-16 number system, used with computers to make binary information easier to interpret by humans. The numbers 0 to 16 are represented by the numerals 0 to 9, plus A, B, C, D, E, and F. An 8-bit byte storing a number from 0 to 255 can readily be represented by the hexadecimal values 0 to FF.

HFS: Hierarchical File System. The file storage system introduced with the MacPlus in 1986, which allows arranging files, applications, and folders in a hierarchy. That is, files can be placed in folders, and folders placed in other folders, just as with a real hardcopy filing system.

High level language: A language that allows representing machine-level operations by mnemonic keywords rather than 1s and 0s. BASIC, COBOL, Pascal, C, and FORTRAN are high level languages.

Highlight: The brightest values in a continuous tone image.

Histogram: A bar-like graph that shows the distribution of gray tones in an image.

HPGL: Hewlett-Packard Graphics Language. Used to define images to be printed with plotters.

HSB color model: A model that defines all possible colors by specifying a particular hue and then adding or subtracting percentages of black or white.

Hue: A pure color. In nature, there is a continuous range of hues.

Icon: A small graphic that represents an object or function on the computer screen.

Image-oriented backup: Any backup system that creates a mirror image of the disk without regard to the files themselves. With such a system, the entire disk must be restored from the backup medium to allow access to the files.

Imagesetter: A high resolution PostScript printer that creates camera-ready pages on paper or film.

IMG: The bit-mapped images produced by programs in the GEM environment.

INIT: A startup document, called an extension under System 7, that becomes part of your operating system when your Mac boots. These often provide some special functions, such as added sound capabilities, extra function keys, font management (Adobe Type Manager is an INIT/extension), or Oscar The Grouch popping up each time you empty the trash. Many scanner drivers are installed as INITs.

Input: Incoming information. Input may be supplied to the computer by the user or to a program by either the user or a data file.

Instructions: The basic set of capabilities of a microprocessor, allowing the chip to load information into a register, move it to another register, increment the data, add or subtract data from a register, and so forth.

Intelligent: Having sufficient programming built-in to carry out certain tasks independently. An intelligent disk drive can accept requests from the operating system, locate the data, and deliver it without detailed instructions on how to do the physical I/O.

Interactive: Allowing user input during run-time.

Interpreter: A program that interprets and carries out each line of another program written in a high level language like BASIC or COBOL. These languages can also be compiled so that DOS can carry out the commands directly. PostScript interpreters for printers perform the same function with the instructions found in PostScript files.

I/O: Input/Output. Used to describe the process whereby information flows to and from the microprocessor or computer through peripherals such as scanners, disk drives, modems, CRT screens, and printers.

Jaggies: Staircasing of lines that are not perfectly

horizontal or vertical. Jaggies are produced when the pixels used to portray a slanted line aren't small enough to be invisible.

Justified: Text that is aligned at both the right and left margins.

K: Kilobyte. In computer terminology, 1024, so that 16K represents 16,384; 64K equals 65,536; 512K corresponds to 524,288; and so on.

Kern: To adjust the amount of space between two adjacent letters.

Knockout: Area on a spot color overlay in which an overlapping color is deleted, so the background color shows through.

Landscape: The orientation of a page in which the longest dimension is horizontal.

Launch: To start a Macintosh application.

Leading: The amount of vertical spacing between lines of text from baseline to baseline.

LHS color correction: A system of color correction based on based on the luminance, hue, and saturation of an image.

Ligature: A combination of two characters squeezed together to form a composite character. Ligatures can confuse OCR programs that use pattern matching until the software has been trained to recognize each ligature combination.

Lighten: An image editing function that is the equivalent to the photographic darkroom technique of dodging. Gray tones in a specific area of an image are gradually changed to lighter values.

Line art: Usually, images that consist only of black and white lines.

Line screen: The resolution or frequency of a halftone screen, expressed in lines per inch. Typical line screens are 53 to 150 lines per inch.

Lines per inch: Abbreviated lpi, lines per inch is the yardstick used to measure halftone resolution.

Lithography: Another name for offset printing, which is a reproduction process in which sheets or continuous webs of material are printed by impressing them with images from ink applied to a rubber blanket on a rotating cylinder from a metal or plastic plate attached to a another cylinder.

Logical: Any feature not physically present but defined anyway for convenience. The physical sectors on a hard disk are arranged contiguously. Logically, they may be arranged in alternating fashion through interleaving.

Luminance: The brightness or intensity of an image. Determined by the amount of gray in a hue, luminance reflects the lightness or darkness of a color. See also Saturation.

LZW compression: A method of compacting TIFF files using the Lempel-Zev compression algorithm. It produces an average compression ratio of 2:1, but larger savings are produced with line art and continuous tone images with large areas of similar tonal values.

Mapping: Assigning colors or grays in an image.

Mask: To cover part of an image so it won't be affected by other operations.

Mass storage: Permanent storage of computer information, usually on magnetic disk but can also include magnetic tape, optical disk, bubble memory, and other non-volatile storage media.

Mechanical: Camera-ready copy with text and art already in position for photographing.

Memory buffer: An area of RAM used to store a file or an image between certain operations, such as printing, storing to disk, or display in an image editing program.

Mezzotint: An engraving that is produced by scraping a roughened surface to produce the effect of gray tones. Image editing and processing software can produce this effect with a process called error diffusion.

Microprocessor: The computer-on-a-chip that is the brains of a personal computer.

Midtones: Parts of an image with tones of an intermediate value, usually in the 25 to 75% range.

Millisecond: One-thousandth of a second.

Moire: In scanning, an objectionable pattern caused by the interference of halftone screens—often produced when you rescan a halftone and a second screen is applied on top of the first.

Monochrome: Having a single color.

Monospaced: Text in which each character takes up exactly the same amount of space. Some OCR programs require specifying that text to be scanned is monospaced. See also proportional spacing.

Motherboard: The main circuitboard of your Macintosh.

Mouse: A pointing device used to indicate an area or point on the screen.

Mount: To activate a floppy or hard disk for use. Disks must be mounted before you can open them.

Multibit: Any scan that uses more than 1 bit to store information about a pixel.

MultiFinder: The Apple system software, used prior to System 7, which makes it possible to have several applications open at once. MultiFinder allows you to move quickly between different programs. Some, like telecommunications software, can run in the background.

Multitasking: The ability of a computer system to handle several different chores simultaneously. Because microcomputers have only one main processor, this is usually done by slicing processor time into individual segments and allowing the programs to share the slices in rotation. DOS is not generally a multitasking operating system, although third-party enhancements can give it these capabilities.

Multiuser: The ability of a computer system to handle several different tasks performed by several different users simultaneously. Unix is the best-known multitasking system among microcomputer users, although it is also available for larger systems.

Negative: A representation of an image in which the tones are reversed. That is, blacks are shown as white, and vice versa.

NTSC: National Television Standard Code, the standard for video in the United States.

NuBus: A system bus used in Mac II series computers, which allows adding high speed expansion cards, such as video adapters.

Object graphics: Vector-oriented graphics, in which mathematical descriptions, rather than bit maps, are used to describe images.

OCR: Optical Character Recognition. The process of converting printed characters into the ASCII characters and other attributes of a bit-mapped image of text.

Offset printing: See lithography.

Omnifont: The ability of an OCR program to recognize most fonts without the need to learn that font – often associated with systems using feature extraction.

Origin: The starting horizontal and vertical reference point for a scan.

Overlay: A sheet laid on top of another to specify spot colors for printing. In programming, a portion of a program that is called into memory as needed, overlaying the previous redundant section of the program. Overlays allow writing programs that are much bigger than those that could fit into memory all at once.

Point: A unit of typographic measurement, approximately 72 to the inch.

Page description language: A programming language that can be used to tell a printer how to handle a given page. PostScript is the most widely known page description language.

Palette: A set of tones or colors available to produce an image.

Pantone Matching System: A registered trade name for a system of color matching. If you tell your printer the PMS number of the color you want, that color can be reproduced exactly by mixing printing inks to a preset formula.

Parallel: To move data several bits at a time, rather than one at a time. Usually, parallel operation involves sending all 8 bits of a byte along eight separate data paths at one time. This is faster than serial movement. Most scanners use parallel connections to move image information.

Parameter: A qualifier that defines more precisely what a program such as Ventura or PageMaker is to do.

Peripheral: Any hardware part of a computer system other than the microprocessor itself and its directly accessible memory. We usually think of peripherals as printers, modems, etc.

Phototypesetting: A process used to expose text and images onto materials that will later be used to produce printing plates. Phototypesetters generally have much higher resolutions than laser printers.

Physical: A feature that exists in reality.

PICT: A graphic image and file format used by the Macintosh and its Clipboard. PICT2 is an enhanced version, which can be used in both 8bit and 24bit formats.

Pixel: A picture element of a screen image; one dot of the collection that makes up an image.

Plate: A thin, light-sensitive sheet, usually of metal or plastic, which is exposed and then processed to develop an image of the page. The plate is put during the printing press to transfer ink or dye to a surface, generally paper.

Plugging: A defect on the final printed page in which areas between dots become filled due to dot gain, producing an area of solid color. See also dot gain.

PMMU: Paged Memory Management Unit. The optional 68851 chip for Motorola 68020-based Macintosh II computers, which provides special memory management needed by Apple UNIX (AUX) and for using virtual memory modes with System 7. Macs using the 68030 and 68040 chips have this memory support built in.

Point: Approximately 1/72 of an inch outside the Macintosh world, exactly 1/72 of an inch within it. Points are used by printers to measure things like type and other vertically oriented objects.

Port: A channel of the computer used for input or output with a peripheral. The serial and parallel ports of the PC are the most widely used.

Portrait: The orientation of a page in which the longest dimension is vertical.

Position stat: A copy of a halftone which can be placed on a mechanical to illustrate positioning and cropping of the image.

Posterization: A photographic effect produced by reducing the number of gray tones in an image to a level at which the tones are shown as bands, as on a poster.

PostScript: Developed by Adobe Systems, PostScript is the most widely used page description language for PCs. It provides a way of telling the printer, typesetter, or imagesetter how to generate a given page.

Prepress: The stages of the reproduction process that precede printing, particularly those in which generate halftones, color separations, and the printing plates themselves are generated.

Preview scan: A preliminary scan that can be used to define the exact area for the final scan. A low resolution image of the full page or scanning area is shown, and a frame of some type used to specify the area to be included in the final scan.

Printer command language: As in Hewlett-Packard Printer Command Language (HPCL). The instructions used to drive HP LaserJet printers and compatibles.

Process camera: A graphic arts camera used to make color separations, photograph original artwork to produce halftones and page negatives, and to perform other photographic enlarging/reducing/duplicating tasks.

Process colors: Cyan, magenta, yellow, and black. The basic ink colors used to produce all the other colors in four-color printing.

Program: Code that instructs the computer how to perform a function.

Proof: A test copy of a printed sheet, which is used as a final check before a long duplication run begins.

Proportionally spaced: Text in which the distance from the center of one character to the center of the next is varied according to the shape and size of the character. The letter i would take up much less space than the letter m, for example. In contrast, monospaced characters all occupy exactly the same amount of space. Some OCR programs require you to specify whether the text is proportional or monospaced. See also monospaced.

Quantization: Another name for posterization.

QuickDraw: A graphics language built into the readonly memory of the Macintosh.

Raster image: An image defined as a set of pixels or dots in row and column format.

Raster image processor: Abbreviated RIP, this is the hardware/software used to process text, graphics, and other page elements into a raster image for output on a printer.

ResEdit: The Apple resource editor, which allows you to modify applications, data files, and system files (if you're brave enough to experiment.)

Read-Only Memory: Memory that can be read by the system but not changed. Abbreviated ROM, read-only memory often contains system programs that help the computer carry out services.

Reflection copy: Original artwork which is viewed and scanned by light reflected from its surface, rather than transmitted through it.

Register: To align images, usually different versions of the same page or sheet. Color separation negatives must be precisely registered to one another to insure that colors overlap in the proper places.

Register marks: Small marks placed on a page to make it possible to align different versions of the page precisely.

Registers: The basic memory locations of a microprocessor, through which all information that is processed passes.

Resolution: The number of pixels or dots per inch in an image, whether it is displayed on the screen or printed.

Retouch: To edit an image, usually to remove flaws or to create a new effect.

RGB color correction: A color correction system based on adjusting the levels of red, green, and blue in an image.

RGB color model: A way of defining all possible colors as percentages of red, green, and blue.

Right-reading image: An image that reads correctly, left to right.

RIP: Raster Image Processor. A device found in printers that converts page images to a format that can be printed by the marking engine of the printer.

RISC: Reduced Instruction Set Computer. A computer system that has a special microprocessor that processes fewer instructions and which, therefore, operates faster. Such systems depend on the software for functions that formerly were handled by the microprocessor.

Saturation: An attribute of a color that describes the degree to which a pure color is diluted with white or gray. A color with low color saturation appears washed out. A highly saturated color is pure and vivid.

Scale: To change the size of a piece of artwork.

Scanner: A device that captures an image of a piece of artwork and converts it to a bit-mapped image that the computer can handle.

Scrapbook: The Apple desk accessory that can be used to keep text and graphics available on a semi-permanent basis. The Scrapbook is a handy tool for scanner users who want to store or move groups of images, since the Clipboard can only hold one at a time.

Screen: The halftone dots used to convert a continuous tone image to a black-and-white pattern that printers and printing presses can handle. Even expanses of tone can also be reproduced by using tint screens that consist of dots that are all the same size (measured in percentages: a 100% screen is completely black).

Screen angle: The alignment of rows of halftone dots, measured from the horizontal (which would be a 0° screen angle).

SCSI: Small Computer Systems Interface. An intelligent interface, used for most scanners in the Macintosh world and for other devices, including hard disk drives.

SCSI ID: The number from 0 to 7 assigned to each device on the SCSI bus. You make this assignment by adjusting a jumper or DIP switch on your equipment, or sometimes, through software. No two devices can have the same ID number, and the higher the number, the greater the priority a device receives. The Mac itself always has SCSI ID 7.

Secondary color: A color produced by mixing two primary colors. For example, mixing red and green primary colors of light produces the secondary color magenta. Mixing the yellow and cyan primary colors of pigment produces blue as a secondary color.

Sector: The smallest section of a track, containing 512 bytes of data.

Separation: See color separation.

Separations: Film transparencies, each representing one of the primary colors (cyan, magenta, and yellow) plus black, used to produce individual printing plates.

Serial: Passing information 1 bit at a time in sequential order. Some scanners use serial connections.

Serif: Short strokes at the ends of letters. Thought to help lead the eye and make text easier to read. Sans serif type lacks these strokes. Serifs can sometimes touch in tightly spaced text, causing problems for OCR software.

Shade: A color with black added.

Shadows: The darkest part of an image, generally with values ranging from 75 to 100%.

Shareware: Software that can be copied and distributed freely for evaluation purposes but which must be registered, usually for a small fee, if you decide to keep using it.

Sharpening: Increasing the apparent sharpness of an image by boosting the contrast between adjacent tones or colors.

SIMM: Single in-line memory module. SIMMs are the small circuit boards used to add memory to Macs and other devices, such as the Laserwriter. Today, SIMMs usually contain 1Mb or 4Mb of memory, but there are other sizes.

Smoothing: To blur the boundaries between tones of an image, often to reduce a rough or jagged appearance.

Solarization: In photography, an effect produced by exposing film to light partially through the developing process. Some of the tones are reversed, generating an interesting effect. In digital photography, the same effect is produced by combining some positive areas of the image with some negative areas.

Source code: The program code generated by a programmer, which may not be directly executable by the computer. If not, it is translated by an assembler or compiler into machine language object code.

Spot: The dots that produce images on an imagesetter or other device.

Spot color: Individual colors used on a page. Usually limited to one or two extra colors besides black to accent some part of a publication.

Spot color overlay: A sheet that shows one of the colors to be used in a publication for a given page. A separate overlay is prepared for each color and all are combined to create the finished page.

Static RAM: Memory that does not need to be refreshed and which, therefore, does not lose its contents when power to the computer is turned off.

String: A series of characters.

Strip: To assemble a finished page by taping or otherwise fastening pieces of film containing halftones, line art, and text together in a complete page negative or positive. The most common format is as a negative, because dirt and other artifacts show up as pinholes which can be easily spotted or opaqued out before the printing plates are made.

Subdirectory: A directory created within another directory, which stores its own separate files.

Substrate: A base substance which is coated with another. In printing, the substrate is generally paper or acetate, and the second substance is usually ink or dye.

Subtractive colors: The primary colors of pigments. When two subtractive colors are added, the result is a darker color which further subtracts from the light reflected by the substrate surface.

Suitcase: A file containing fonts or desk accessories, so named because its icon looks like a small suitcase.

System software: Your Mac's operating system, which includes the System file and Finder, as well as other components.

System file: The file used to start up your Macintosh, and regulate the transfer of information among all the other components of the system software.

System level interface: An interface over which information is passed in logical form.

TARGA: Truevision Advanced Raster Graphics Adapter. A type of video board pioneered by Truevision, which produces files compatible with NTSC signals.

Template: A publication that is used as a framework to provide the basic structure and layout for a publication.

Terminator: A device that absorbs signals at the end of a bus, preventing electronic "bounce-back." Your SCSI bus must have two terminators, one at the first and last devices. Some devices are internally terminated. Others require an add-on device.

Text file: Usually an ASCII file, often created by selecting Save Text Only from within an application.

Threshold: A predefined level used by the scanner to determine whether a pixel will be represented as black or white.

Thumbnail: A miniature copy of a page or image, which gives you some idea of what the original looks like without having to open the original file or view the full size image.

TIFF: Tagged Image File Format. A standard graphics file format which can be used to store gray scale and color images.

Tint: A color with white added to it. In graphic arts, often refers to the percentage of one color added to another.

Toner: A pigmented substance used in page printers (and office copiers) to produce an image on a page.

Trapping: The ability of an ink to transfer as well onto another layer of ink as to the bare paper itself. In halftoning, poor trapping will result in tonal changes in the final image.

Triad: Three colors located approximately equidistance from one another on the color wheel. Red, green, and blue make up a triad; cyan, magenta, and yellow make up another. However, any three colors arranged similarly around the wheel can make up a triad.

True Color: A system in which any pixel in the image can be any of the 16.8 million colors available in a 24-bit color mode. This is in contrast to systems that also access the full 16.8 million color gamut but limit a given image to a smaller palette of colors chosen from the larger range. For example, you may be able to use only 256 colors even though any of the millions available can be selected for that palette.

Type code: A four-letter code that tells the Macintosh what kind of document a file is. It is used with the creator code, which represents the particular application that created the file. Some applications can create several different types of documents.

Undercolor removal: A technique that reduces the amount of cyan, magenta, and yellow in black and neutral shadows by replacing them with an equivalent amount of black. It can compensate for trapping problems in dark areas. See also gray component removal.

Unfragmented: A hard disk that has most of its files stored in consecutive sectors, rather than spread out over the disk. Such an arrangement allows more efficient reading of data, requiring less time to move the read/write head to gather the information.

Utility: A program that performs some useful system or maintenance function, as opposed to an applications program.

Vector image: An image defined as a series of straight line vectors. The beginning and ending points of each line are stored and later adjusted as the image is sized.

Vignette: In prepress terminology, an image with a continuous gradation of tones.

Virtual disk: An electronic, or RAM, disk created in memory to mimic a real disk drive—only much faster.

WMF: Windows Metafile. A vector-based graphic file format for IBM PC-based computers.

WORM: Write Once Read Many (or Mostly). Optical disk technology that allows writing to the disk by the user, although a given section cannot be erased and re-used.

Wrong-reading image: An image that is backward relative to the original subject—that is, a mirror image.

X-height: The height of a lowercase letter, excluding ascenders and descenders.

Zoom: To enlarge part of an image so that it fills the screen, making it easier to work with that portion.

Appendix B

What Is Shareware?

Several years ago, I came to the conclusion that many software vendors would do just as well giving away the initial copies of their software and charging only for upgrades. After all, over the long run, upgrades can end up costing you more than the original software anyway. Making the original copies free would provide a broader base of users to pay for those lucrative upgrades.

As a writer in the computer industry, I receive just about any software package I want free of charge. For example, in preparing my first scanner book, I acquired copies of all five of the add-on PostScript interpreters available at that time, a dozen or more image editing programs, and all the leading OCR software.

While getting software this way allows me to do in-depth analyses for my book projects, remaining current with the state of the art is a bit more complicated. I usually have to send out seven or eight dozen letters annually asking for upgrades of specific packages. Some vendors allow authors to register the copies they receive (Joining the ranks of "regular" users is a good way to learn how good customer support is) and I do this whenever possible. Of course, that gets me on the list for upgrade offers at $35 to $99 or so per upgrade.

I end up paying for these upgrades in many cases, even though I could write to the vendor for another one-shot copy of the product. Even if I don't use the software regularly in my work, paid upgrades are useful to me because most vendors are very prompt about responding and may even inform registered users before the upgrade news reaches the computer press. Some of my initially free software packages have cost me $200 or more in repeated upgrades.

When I notice that, I realized that the upgrade process made a lot of sense. Truly valuable software continues to be valuable, and users want to preserve their investments while gaining enhanced features (and bug fixes) for continual, affordable fees.

Wouldn't it be nice if software were provided to you free initially, followed by a regular program of upgrades for relatively low fees? You wouldn't mind spreading out the payments for software that allowed you to try-before-you-buy at little risk, would you?

Actually, there is a classification of software with exactly those features. It's called shareware. Shareware has been with us since the late Andrew Fluegelman introduced the communications program PC-Talk in 1982. He asked users who found it helpful to send contributions.

Exactly What Is Shareware?

I explained the shareware concept briefly in the Introduction to this book. In summary, such programs resemble their traditionally marketed counterparts in many ways. Both are often (but not always) developed by teams of professional programmers. It is difficult to discern shareware from conventional retail software simply on the basis of features, functionality, user interface, or overall quality. Shareware is just commercial software that is distributed on a try-before-you-buy basis.

When you stop to think about it, this honor system is not much different from the way retail software is sometimes handled. True, many users don't pay for the shareware copies of software that they use regularly. But then, neither do a large number of users of retail software. Industry pundits have pointed to the huge number of pirated copies of WordStar that were distributed in the early 1980's as a key reason for its commercial success.

People were passing around thousands of copies of the program, helping WordStar entrench itself in many organizations. A significant number of these companies and individuals eventually "went legitimate" by purchasing copies. I can't endorse pirating software as a viable marketing scheme for packages sold through retail channels. However, in the real world, that's what sometimes happens.

Millions of people are trying before they buy—on the sly—with retail software. The difference with shareware is that the practice is above board, and nobody gets stuck with expensive software that doesn't fit his or her needs.

Shareware happens to be upgraded more frequently than most commercially distributed software. The upgrade fees are lower, too. You may get a free upgrade or two with your initial registration and then pay a very nominal fee for additional upgrades as the software is enhanced further. As I noted in the beginning of this Appendix, paid upgrades can be a valuable way to remain current, even if you eventually end up paying more than you did for the original package.

In the true spirit of sharing, this appendix borrows—with permission—from some of the leading thinkers in the field. Special thanks go to Rob Rosenberger, author of a useful book called *Shareware: Try Before You Buy.* Many of the ideas in the following sections are Rob's and are paraphrased from his book. If you'd like to know more about shareware, look in Appendix C where you will find instructions and an address for ordering the expanded third edition of Rob's guide. He reprints the guidelines of the Association of Shareware Professionals (ASP), a pioneering group that has helped establish useful standards for shareware.

No Commercial Potential?

I've tried to emphasize that shareware is commercial software. Its authors have the same intent to make money from their efforts as those who write retail programs do. Shareware is not freeware—a term coined by Fluegelman and registered as a trademark (when capitalized) by Headlands Press. Freeware programs are copyrighted (or in some cases patented) but can generally be used without giving the authors compensation. Because they are copyrighted works, you can not distribute altered copies of freeware programs without permission. Nor can you sell freeware as if it belonged to you, whether or not you have made any changes in the program. Freeware code remains the property of the authors. You can use it at no cost, but that's about all.

Shareware is also often confused with public domain software. Public domain programs are open to use by the public without any compensation to the authors for just about any purpose. There are no copyrights or patents restricting your use of public domain software. You can incorporate public domain code into your own products and market it, give it away, or do anything else you like. However, so can anyone else. Public domain software is so widely distributed and used that it really has little value – other than for its intended use. PD software can range from poor to excellent in quality, but you must be cautious. There is little assurance that your copy hasn't been "improved" by some well meaning (or evilly motivated) programmer.

A Test Drive

Okay, now we know that shareware is not the same as freeware or public domain software. What else makes shareware different from retail software?

Unlike retail programs, shareware offers you a chance to test drive the software before you pay for it. In most cases, you can't return retail software after the package has been opened. With shareware, you don't have that problem. You are granted anywhere from 10 to 30 days to use a fully functional version of the program before deciding if you want to continue to use it.

If you don't want to make the shareware package a regular part of your software repertoire, you can erase it from your hard disk or return the floppy disk to your library. Better yet, you can pass along a copy to a friend, who might like to evaluate the program.

Despite the low fees, shareware authors can make as much or more per copy than they would from a retail version of the program. That's because without the advertising, manufacturing, and in some cases, support costs of retail software, small shareware fees can still contain a reasonable profit.

When you send a fee to the shareware author, you receive a license to continue using the software. Often you get some other goodies, such as a bound copy of the manual or free updates. In addition, a registration fee sometimes frees you from a beep or message in the shareware copy of the software.

Advantages of Registration

I'll be pretty explicit about the advantages of registering shareware, because this important step is what makes or breaks a program. I'll start with the most direct benefits.

- *Legal right to continue using the software beyond the trial period.*
- *Extra goodies.* As I described above, you can get a manual, free update, or some other incentive when you register.
- *Support by telephone or bulletin board.* Few shareware authors have time to talk personally with all the users of their packages, but many will speak with you by telephone if you have a serious problem that can't be solved by a thorough reading of the manual. They also install their own bulletin board systems on which you can post questions, share tips with other users, and download utilities and other useful files.
- *Commissions.* Some programs offer commissions to registered users. If anyone registers a copy made from your registered copy, you receive a cut, typically $10 or so. Commissions are hardly the best reason to register software, but they can be an important bonus for some.
- *Source code and site licenses.* When you register, you can also, under certain circumstances, receive the source code. This can be important for larger businesses which may want to customize software for their own needs. It's more likely in the case of applications programs such as accounting software. Registration can also open the door to site licensing, which gives you permission to use a large number of copies within an organization for a reduced per copy fee.

- *You get on their mailing list.* Your shareware author will sometimes let you know first about upgrades, new products the company has, and other things you are likely to find of interest.
- *Continued access to decent software at reasonable prices.* If you like a program enough to consider registering it, you probably will be interested in having other programs of similar quality available to you in the future. Only by paying your fair share can you be assured of that. Don't be tempted to let the other guy subsidize your usage or to postpone registering your software because you're busy.

What Shareware Is Not

Aside from these distinctions, there are some other criteria that determine whether or not a package should rightly be considered shareware.

Shareware is not "crippleware," that is, software with one or more major functions disabled. The demo version of some applications don't let you access certain features, such as Help, and the size of the files you can save is limited. Shareware can do everything that the registered version does and has the same complete documentation (even if it is only a TeachText or MacWrite file on a disk).

Shareware is not demoware, which is software that shows you what a package does without letting you try it for yourself. Such software is little more than a computerized sales pitch, offering limited opportunity for user input.

Shareware is not an old or outmoded version of the software, which you can use and distribute. The author can't require you to pay for an up-to-date release. Nor can shareware be software that is fully functional for a limited period of time, after which it self-destructs, locks up, or becomes non-functional.

In short, shareware can't be a database that accepts only 100 records or a word processor that won't save documents. It can't be a program that stops working unless you send money.

In his book, Rob Rosenberger lists some common misconceptions about shareware. For example:

- *Shareware is second-rate software.*

  And so is much of the retail software you buy – and can't get a refund on. The distribution method has little bearing on the quality of the package.

- *Shareware authors aren't successful enough to make it in the retail business.*

  At least five companies report sales of more than $1 million in shareware each year.

- *You can't rely on shareware authors to support their products.*

  Many of them do a better job on this count for individuals than Apple, and some even extend support to non-registered users (who are called "potential customers"). I've spent hours waiting in telephone queues for support from major software vendors but was able to call a shareware author at home when I had a major problem.

- *Shareware never wins awards for being great.* Has anybody counted up the honors programs like STUFFIT have received?

- *Shareware is cheap; you get what you pay for.*

  Actually, you don't pay for what you don't get, like fancy packaging you throw away, advertising you ignore, and corporate jets for a president who wouldn't know an op code if it bit him.

- *Shareware spreads computer viruses.*

  I've actually seen a computer virus at work only twice. The first virus infected my Macintosh, and was easily traced back to a copy of Aldus Freehand I had received directly from Aldus. After that, I installed an INIT to check each disk I inserted into my system. That program (itself shareware) spotted my *second* virus.

  In the second instance, a user who was having trouble with an application sent along a copy of the application for me to test. His disk was had the WDEF virus all over it, and never made it past my INIT. On checking further, it turned out this user hadn't actually purchased this software, nor most of the other programs he used. His virus infection had come not from shareware but from pirated software.

In my work, I probably test more shareware in a year than you will encounter in a lifetime, and I've never had a problem. BBS operators, information services, and shareware mail order companies are fairly diligent at policing the files they distribute. I wouldn't lose any sleep worrying about viruses from any source at this time.

Obtaining Shareware

There are many shareware programs that are potentially of interest to scanner users. This section will tell you how to get the latest versions.

- Write the author at the address provided with any copy of the program you might have. This is probably the best choice if you've already decided to register your copy. Register, and he'll send you a copy of the very latest version.

  If you're not yet ready to register, you'll want to explore one of the following options:

- If you have a modem, the quickest way is to download the software from a computer bulletin board system (BBS). A local bulletin board has the advantage of allowing you to get shareware for free, if the board doesn't charge a user fee and the phone call is toll-free for you.

  Some BBS systems are specialized, serving only certain computer systems or interests, such as desktop publishing. Nearly any general interest, or Macintosh-oriented BBS will have a copy of many shareware packages. If you're not familiar with the BBSs in your area, a local computer user group, high school computer club, or the staff at a friendly computer store can help you out.

  If you can't find the shareware you want on a local board, purchase a copy of *Computer Shopper*. Each monthly issue of this massive tabloid includes a list of the major bulletin board systems and user groups, arranged geographically.

  The BBS method does not guarantee you will get the absolute latest version of shareware, because many BBSs depend on their users to upload the software. In larger cities, the most frequently accessed BBSs will have very recent copies of popular shareware and public domain programs.

- You can also obtain the very latest version of most shareware from some of the online information services, such as the CompuServe Information Service (CIS). Shareware on these major networks is often uploaded by the authors themselves, so you can be assured of getting a complete copy only days after its release.

  The only disadvantage to downloading shareware from the information services is that the service isn't free. CompuServe, for example, charges $6 per hour for 300 baud connect time and $12 per hour for 1200 or 2400 baud service. That's only the basic fee to use the information service's computer. You also have to pay for whatever telecommunications network you use to reach CompuServe's computer system. That network surcharge can range from 25 cents an hour to $2 an hour or more. The exact charge to use the network depends on time of day ("office" hours are prime time; evenings are off-peak on some services) and whether you link up with CompuServe through one of their own network nodes (conveniently located in cities throughout the country) or through a public network like Tymnet or Telenet. Most services also charge a sign-up fee.

  Logging onto CIS, finding shareware, downloading it, and reading some of the interesting messages from other users might take you an hour or so. However, the $6.25 to $15 it will cost will be well spent. You'll get the latest version and access to another source of information about quality shareware. Remember that although you pay for the connect time, you still receive the shareware version, so you will need to register your copy if you decide to continue using it.

  When you download shareware from an information service or BBS, it will probably be in compressed, or archived, format. You'll need a program like STUFFIT to reconstruct the file. These can be downloaded from the same source or obtained from one of the outlets described below.

- User groups are another source of shareware. *Computer Shopper* can lead you to some local groups, many of which maintain shareware libraries of disks that users can borrow and copy. You may not even have to join, or you may be able to get a friend who is a member to borrow shareware for you. Again, local computer stores can help you locate user groups in your area.
- Many smaller computer stores and even some book stores that carry computer book titles "sell" shareware disks. I've purchased shareware disks from a Book Warehouse outlet in Rochester, NY, and from a struggling computer store here in my home town (population 11,000). There is a restriction, (suggested by the Association of Shareware Professionals) on how much such organizations can charge for shareware disks, so you'll usually see these disks priced at $8 or less. The store makes a small profit (their costs are more than the 19 cents they paid for the disk; remember, commercial operations have overhead). The shareware gets wider distribution. It's a good system.
- There are dozens of public domain/shareware mail order firms that manage to eke out enough profits from selling software to pay for toll-free 800 numbers, computerized disk duplicating systems, and floor space on which their operators can stand by. Expect to pay from $2 to $6 for each disk. The quality of the product varies all over the place. Some of the low price firms provide only one or two programs per disk for your two bucks. Others pack them full of compressed software for the same price. Just remember that if you deal with an unknown firm, you have no way of knowing whether or not you're getting the latest version.

Summary

Shareware is commercial software that is distributed through a try-before-you-buy program. If you like the package, you can register it and receive a number of valuable benefits. Organizations like the Association of Shareware Professionals (ASP) help insure that you receive maximum value for the lowest price.

There are many shareware programs besides PRO-CR of interest to scanner users. To obtain shareware you can use one of these sources:

- Write the author.
- If you have a modem, the quickest way is to download the software from a computer bulletin board system (BBS).
- You can also obtain the very latest version of most shareware from some of the online information services, such as the CompuServe Information Service (CIS) and GEnie.
- User groups are another source of shareware
- Many smaller computer stores and even some book stores that carry computer book titles "sell" shareware disks.
- There are dozens of public domain/shareware mail order firms that manage to eke out enough profits from selling software to pay for toll-free 800 numbers and other necessities.

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 99 HOMEWOOD, IL

POSTAGE WILL BE PAID BY ADDRESSEE

BUSINESS ONE IRWIN
1818 Ridge Road
Homewood, IL 60430-9924

ATTENTION
Joan Zagone

BUSINESS ONE IRWIN is dedicated to providing you with the most timely, authoritative, and comprehensive computer guides to meet your personal and professional needs. Our goal is to offer you excellent reference tools written by experienced, highly-qualified authors. Please take a moment to answer the following questions so that we can continue to develop our products based on your needs.

Thank you.
BUSINESS ONE IRWIN

Current Computer Hardware:

- ☐ IBM or compatible
- ☐ Apple or Macintosh
- ☐ Commodore
- ☐ Atari

Disk Density:

- ☐ High
- ☐ Low

Disk Size:

- ☐ 5.25"
- ☐ 3.5"

Is It Portable?

- ☐ Yes
- ☐ No

Printer Type:

- ☐ LaserJet
- ☐ PostScript
- ☐ Dot Matrix
- ☐ Other

Peripherals:

- ☐ Modem
- ☐ Scanner
- ☐ Mouse
- ☐ Networking
- ☐ FAX
- ☐ Other *(please specify)* ____________________

Operating System:

- ☐ MS-DOS
- ☐ OS/2
- ☐ Macintosh
- ☐ UNIX

Programming Languages:

- ☐ BASIC
- ☐ C
- ☐ Pascal
- ☐ Prolog
- ☐ Assembly
- ☐ HyperTalk

The Most Recent Software Program I Purchased Is: ____________________

Software I Use Most Often:
(please check all that apply and state name of product)

- ☐ Word Processing____________________
- ☐ Spreadsheet____________________
- ☐ Database/File Mgmt____________________
- ☐ Desktop Publishing____________________
- ☐ Presentation Graphics____________________
- ☐ Accounting____________________
- ☐ Statistical Analysis____________________
- ☐ Financial/Tax Planning____________________
- ☐ Integrated ____________________
- ☐ Business Form Design ____________________
- ☐ Utilities____________________

Additional Comments: ____________________

Name____________________
Title ____________________
Organization____________________
Address ____________________
City/State/Zip____________________
Telephone____________________
☐ Please send a catalog of Business One Irwin books.

J15-3448-01-01

businessOne
IRWIN
1818 Ridge Road, Homewood, IL 60430

QUALITY PUBLISHING FOR WORLDWIDE BUSINESS MARKETS

Appendix C

Resources

This appendix lists the addresses and phone numbers (where I had them) of every supplier I could find in several key categories of interest to scanner users. There are no product descriptions or reviews included. For comparisons, I recommend *The Complete Scanner Handbook for Desktop Publishing, Macintosh Edition,* or recent issues of *MacWorld, MacUser, and InfoWorld.* I've listed the names of key products for each company, except in the scanner section, since scanner models change relatively frequently.

Scanners

Abaton
48431 Milmont Dr.
Fremont, CA 94538
800-821-0806
415-683-2226

Advanced Vision Research, Inc
2201 Qume Dr
San Jose, CA 95131
408-434-1115

Aedex Corp
1070 Ortega Way
Placentia, CA 92670
714-632-7000

American Microsystems
2190A Regal Pkwy
Euless, TX 76040
800-648-4452
817-571-9015

Asuka Technologies, Inc.
17145 Von Karman Ave.
Suite 110
Irvine, CA 92714
714-757-1212

Atex Magazine Publishing Systems
32 Wiggins Ave
Bedford, MA 01730
617-276-7195

Barneyscan Corp
1125 Atlantic Ave
Alameda, CA 94501
415-521-3388

Black Box Corp
P.O. Box 12800
Pittsburgh, PA 15241
412-746-5500

C.P. (USA), Inc.
One World Trade Center
Suite 2841
New York, NY 10048
212-839-0902

Caere Corp.
100 Cooper Ct.
Los Catos, CA 95030
408-395-7000

Canon U.S.A., Inc
One Canon Plaza
Lake Success, NY 11042
516-488-6700

Chicony America Inc
1641 W. Collins Ave
Orange, CA 92667

Chinon America, Inc.
Information Equipment Division
660 Maple Avenue
Torrance, CA 90503
213-533-0274

CompuScan Inc.
300 Broadacres Dr
Bloomfield, NJ 07003

Comteq USA
13949 Equitable Rd
Cerritos, CA 90701
800-521-4892
213-404-1619

CyberChrome, Inc
25 Business Park
Branford, CT 06405
203-483-8815

Desktop Technology Corp.
986 Mangrove, Suite B
Sunnyvale, CA 94086
800-759-4001

Dest Corp
1015 E. Brokaw Rd
San Jose, CA 95131
408-436-2700

DFI, Inc.
2544 Port St.
West Sacramento, CA 95691
916-373-1234

ECA C&C Products, Inc.
38 Route 46 East
Lodi, NJ 07644
800-442-6872
201-478-0302

ECRM
554 Clark Rd
Tewksbury, MA 01876
508-851-0207

Esco Systems, Inc
540 Weddell Dr. #2
Sunnyvale, CA 94089
408-744-9001

Focus Technology
18226 W. McDurmott
Irvine, CA 92714
800-852-0105
714-553-8626

Fujitsu America, Inc
3055 Orchard Dr
San Jose, CA 95134
800-626-4686
408-432-1300

GUIS America, Inc.
3675 Placentia Ct.
Chino, CA 91710
714-590-0801

Hewlett-Packard Co
19310 Pruneridge Ave
Cupertino, CA 95104
800-752-0900

Houston Instruments
8500 Cameron Rd
Austin, TX 78753
800-444-3425
512-835-0900

Howtek, Inc.
21 Park Ave
Hudson, NH 03051
603-882-5200

KYE International Corp.
12675 Colony St.
Chino, CA 91710
714-590-3940

Kyocera Unison, Inc
1321 Harbor Bay Pkwy
Alameda, CA 94501
800-367-7437
415-748-6680

Logitech, Inc.
6505 Kaiser Dr.
Fremont, CA 94555
415-795-8500

Marstek, Inc
17795-F Skypark Cir.
Irvine, CA 92714
714-833-7740

Microtek Lab, Inc
680 Knox St
Torrance, CA 90502
800-654-4160
213-321-2121

Migraph, Inc.
200 S. 333rd St
#220
Federal Way, WA 98003
800-223-3729
206-838-4677

Mitsubishi Electronics
America, Inc.
991 Knox St.
Torrance, CA 90502
213-515-3993

NCL America, Inc.
1753 S. Main St.
Milpitas, CA 95035
408-956-1040

Nisca, Inc.
1919 Old Denton Rd.
#104
Carrollton, TX 75006
214-245-0942

Prime Option, Inc.
2341 W 205th St.
Torrance, CA 90501
213-618-0274

UMAX Technologies, Inc
2352 Walsh Ave
Santa Clara, CA 95051
800-562-0311
408-982-0771

Varityper
11 Mt. Pleasant Ave
East Hanover, NJ 07936
800-631-8134
201-887-8000

Xerox Imaging Systems
185 Albany St
Cambridge, MA 02139
617-864-4700

Xerox Imaging Systems, Inc
535 Oakmead Pkwy
Sunnyvale, CA 94086
408-245-7900

OCR

Advanced Vision Research
2201 Qume Dr
San Jose, CA 95131
Tiger-EMS MegaScan

Advanced Vision Research, Inc
2201 Qume Dr
San Jose, CA 95131
408-434-1115
MegaRead

Advanced Vision Research, Inc.
2201 Qume Dr
San Jose, CA 95131
408-434-1115
TigerOCR

Caere Corp
100 Cooper Ct
Los Gatos, CA 95030
408-395-7000
OmniPage

Calera Recognition Systems, Inc
2500 Augustine Dr
Santa Clara, CA 95054
800-544-7051
408-986-8006
WordScan

CompuScan, Inc
1373-20 Veterans Memorial Hwy
Hauppauge, NY 11788
516-361-7475
TypeReader

CTA, Inc
747 Third Ave
3rd Flr
New York, NY 10017
800-252-1442
212-935-2280
TextPert series

Datacap, Inc
P.O. Box 5048
Evanston, IL 60204
Paper Keyboard

Dest Corp
1015 E. Brokaw Rd
San Jose, CA 95131
408-436-2700
Recognize
Text Pac. 3.0

Inovatic, Inc
1911 N. Fort Myer Dr. #708,
Arlington VA 22209
703-522-3053
Readstar Series

Logitech, Inc
6505 Kaiser Dr
Fremont, CA 94555
415-795-8500
CatchWord Intelligent OCR

M&P Computers & Communications
1270 Oakmead Pkwy. #111
Sunnyvale, CA 94086
408-738-8578
ReadPro

Nisca, Inc
1919 Old Denton Rd. #104
Carrollton TX 75006
800-245-7226
214-242-9696
Niscan OCR

OCR Systems, Inc
1800 Byberry Rd. #1405
Huntingdon Valley PA 19006
800-233-4627
215-938-7460
ReadRight

Ocron, Inc
3350 Scott Blvd
Bldg. 36
Santa Clara, CA 95054
ReadPro OCR software

Olduvai Corp
7520 Red Rd
Suite A
South Miami, FL 33143
800-822-0772
305-665-4665
Read-It

Pattern Analytics
11 Tucker St
Raleigh, NC 27603
800-762-5542
919-828-5100
Carets/By Hand

ReadPro OCR Software
Ocron, Inc
3350 Scott Blvd
Bldg. 36
Santa Clara, CA 95054
800-933-1399
OmniPlus

Prism Enterprises, Inc.
14703-E Baltimore Ave.
Suite 248
Laurel, MD 20707
TextScan

Soricon Corp
4725 Walnut St
Boulder, CO 80301
800-541-7226
303-440-2800
Datasweep 2

Videx, Inc
1105 N.E. Circle Blvd
Corvallis, OR 97330
503-758-0521
TimeWand series

Image-scanning and Editing Software

Adobe Systems, Inc
1585 Charleston Rd
Mountain View, CA 94039
800-344-8335
415-961-4400
Photoshop, Adobe Illustrator Streamline

Aldus Corp.
411 First Ave. S
Suite 200
Seattle, WA 98104
206-622-5500
Freehand

Astral Developmenl Corp
Londonderry Square #112,
Londonderry, NH 03053
603-432-6800
Picture Publisher

Computer Aided Technology, Inc
7411 Hines Pl. #212,
Dallas, TX 75235
214-631-6688.
CAT Image Enhanoer

Delta Tao Software, Inc.
Sunnyvale, CA
408-730-9336
Color MacCheese

Deneba Software
7855 NW 12th St. Suite 202
Miami, FL 33126
305-594-6965
Canvas, UltraPaint

Electronic Arts
1820 Gateway Dr
San Mateo, CA 94404
Studio/1, Studio/8, Studio/32

Hammerlab Corp
254 College St. #503
New Haven, CT 06510
800-351-4500
203-624-0000
ScanDo

LetraSet USA
40 Eisenhower Dr.
Paramus, NJ 07653
800-634-3463
ImageStudio

Silicon Beach Software
9770 Carroll Center Rd.
San Diego, CA 92126
619-695-6956
SuperPaint, Digital Darkroom

SuperMac Technologies
295 N. Bernardo Ave.
Mountain View, CA 94043
415-962-2900
PixelPaint Professional

Color Separation Software

Aldus Corp.
411 First Ave. S
Suite 200
Seattle, WA 98104
206-622-5500
Aldus Pre-Print

Radius, Inc
1710 Fortune Dr
San Jose, CA 95131
800-527-1950
408-434-1010
PrecisionColor Calibrator

Video Digitizers

Data Translation, Inc
100 Locke Dr
Marlboro, MA 01752
508-481-3700
ColorCapture

Digital Vision, Inc
270 Bridge St
Dedham, MA 02026
800-346-0090
617-329-5400
ComputerEyes B/W
ComputerEyes/Pro Color

Nikon Electronic Imaging
623 Stewart Ave
Garden City, NY 11530
516-222-0200
HQ-1500C HDTV Still
Camera System

Redlake Corp
Truevision, Inc
7351 Shadeland Station
Suite 100
Indianapolis, IN 46256
Targa 16, TIPS Imaging Software

Imagesetters

Agfa Compugraphic
200 Ballardvale St
Wilmington, MA 01887
800-227-2780
508-658-5600

Birmy Graphics Corp
244 NW 21 Terrace
Miami, FL 33142
305-633-3321
Linotronic Co.
425 Oser Ave
Hauppage, NY 11788
800-633-1900
416-567-7111

Monotype, Inc
2500 Brickvale Dr
Elk Grove, IL 60007
708-350-5600

Optronics
7 Stuart Rd
Chelmsford, MA 01824
508-256-1872

Varityper
11 Mt. Pleasant Ave
East Hanover, NJ 07936
800-631-8134
201-887-8000

Monitors:

Apple Computer, Inc.
20525 Mariani Ave.
Cupertino, CA 95014
408-996-1010
Apple monitors

Cornerstone Technology
1990 Concourse Dr.
San Jose, CA 95131
408-435-8900
SinglePage series

Micro Display Systems
PO Box 455
Hastings, MN 55033
800-328-9524
612-437-2233
Genius series

Moniterm
5740 Green Circle Dr
Minnetonka, MN 55343
Viking monitors

Nanao USA Corp.
23510 Telo Ave. #5
Torrance, CA. 90505
213-325-5202
FlexScan

NEC Technologies
1414 Massachusetts Ave
Boxborough, MA 01719
508-264-2860
MultiSync series

Relisys
320 S. Milpitas Blvd.
Milpitas, CA 95051
408-945-9000
RE-1520

RasterOps Corporation
2500 Walsh Ave.
Santa Clara, CA 95051
408-562-4200
RasterOps monitors

Sampo America
5550 Peachtree Industrial Blvd
Norcross, GA 30071
Sampo Office Pro II and IIa monitors

Samsung Information Systems
3655 N 1st St.
San Jose, CA 95134
MP PageMaster

Sigma Designs, Inc
46501 Landing Pkwy
Fremont, CA 94538
415-770-0100
Silverview displays

Sony Corporation of America
1 Sony Dr.
Park Ridge, NJ 07656
201-930-7071
CDP-1304 Multiscan

SuperMac Technologies
295 N. Bernardo Ave.
Mountain View, CA 94043
415-962-2900
SuperMac monitors

Tatung Co. of America, Inc.
2850 El Presidio St.
Long Beach, CA 90810
800-827-2850
213-979-7055

Taxan U.S.A. Corp.
161 Nortech Pkwy.
San Jose, CA 95134
800-829-2641
408-946-3400

Vermont Microsystems, Inc.
11 Tigan St.
Winooski VT 05404
800-354-0055
802-655-2860
Page Manager Series

Wyse Technology, Inc
3471 N. First St
San Jose, CA 95134
800-438-9973
408-473-1200
AM-4322

Desktop Publishing Software:

Aldus Corp
411 First Ave. S
Suite 200
Seattle, WA 98104
PageMaker

LetraSet USA
40 Eisenhower Dr.
Paramus, NJ 07653
800-634-3463
Ready,Set,Go
DesignStudio

Quark, Inc.
300 S. Jackson, Suite 100
Denver, CO 80209
800-543-7711
QuarkXpress

Xerox Desktop Software
9745 Business Park Ave
San Diego, CA 92131
Ventura Publisher

Miscellaneous Products

CompuServe Information Service
Box 20212
Columbis, OH 43220
800-848-8199
On-line services

Iomega Corp
1821 West 4000 South
Roy, UT 84067
801-778-1000
Bernoulli Box II

Pantone Press, Inc
55 Knickerbocker Rd
Moonachie, NJ 07074
Pantone Matching System

Xionics, Inc.
765 The City Drive
Suite 340
Orange, CA 92668
714-971-4717
Image processing engine

Kennect Technology
271 E. Hacienda Ave.
Campbell, CA 95008
800-552-1232
Drive 2.4, Rapport

PostScript Compatible Interpreters

Adobe Systems, Inc
1585 Charleston Rd
Mountain View, CA 94039
Adobe PostScript Cartridge

Custom Applications, Inc
900 Technology Park Dr
Bldg. 8
Billerica, MA 01821
508-667-8585
Freedom of Press

Laser Connection, Inc
7851 Schillinger Park W
Mobile, AL 36608
UltraScript

Conversion and Utility Software

Adobe Systems, Inc
1585 Charleston Rd
Mountain View, CA 94039
800-344-8335
415-961-4400
Adobe Streamline

Central Point Software, Inc.
15220 NW Greenbrier Pkwy
#200
Beaverton, OR 97006
503-690-8090
Copy II

Dataviz, Inc.
35 Corporate Drive
Trumbull, CT 06611
203-268-0030
MacLink Plus

Dayna Communications, Inc.
50 S. Main St.
Salt Lake City, UT 84114
801-531-0203
DOS Mounter

Insignia Solutions, Inc.
526 Clyde Avenue
Mountain View, CA 94043
415-694-7600
SoftPC
AccessPC

Fifth Generation Systems
10049 N. Reiger Rd.
Baton Rouge, LA 70809
504-291-7221
Fastback II
Suitcase II

Wildflower Software
21W171 Coronet Rd.
Lombard, IL 60148
708-916-9360
SnapJot screen capture software

Color Filters

Rosco
36 Bush Ave.
Port Chester, NY 10573
914-937-1300

Rosco
1135 N. Highland Ave.
Hollywood, CA 90038
213-462-2233

Books

ShareWare "Try Before You Buy"
I described this book on the Share Ware concept in Appendix B. You can order your own copy for $4.95 plus shipping and handling from:

Barn Owl Software
P.O. Box 74
O'Fallon, IL 62269

Appendix D

Ordering Information

If you liked the software provided with this book, you may want to order the full or enhanced versions available of each package. Readers of this book can get some valuable applications at a fraction of their true worth. Studio/1 had an original list price of $150. You can order it directly from Electronic Arts for a special price of just $55.00. Similarly, QuickGIF and PCXtc can be upgraded for just $30 and $10 respectively.

Use the coupons on the following pages, or write directly to the vendors at the addresses shown.

Yes, please send me Studio/1 1.0 at this great price!

Ordering by phone

To order by phone with VISA/MC, call 800-245-4525 anytime from the United States or Canada. Outside North America, call 415-571-7171, extension 555 from 8 a.m. to 5 p.m. Pacific Standard Time, Monday through Friday.

Ordering by mail

Simply send in the order form below along with your payment (U.S. dollars, U.S. or Canadian bank, VISA/MC.) When computing your total amount due, add shipping and handling charges as follows: $3.50 for first item, $1 for each additional item. Residents of these states please add sales tax: CA: 7%, CT: 8%, MA: 5%, IL: 6.75%,TX : 8.25%. Please allow 2 to 4 weeks for delivery

Visa __ MC __ Account # ____________________ __________

Expiration Date / Signature ____________________ __________

Name ____________________ __________

Address ____________________ __________

City ____________________ __________

State/Province ____________________ __________

Zip/Postal Code ____________________ __________

Daytime Phone ( ) ____________________ __________

Studio/1	Quantity ____	Special Price :	$55.00
		Sales tax:	$______
		Shipping/Handling:	$______
		TOTAL:	$______

Mail to: **Electronic Arts**
PO Box 7530
San Mateo, CA 94403

Regular Price is $79.00

Please send me the latest version of QuickGIF and QuickGIF Plus!

Name ______________________________

Address ______________________________

City ______________________________

State/Province ______________________________

Zip/Postal Code ______________________________

Price: $30.00

California residents add sales tax.

Mail to: **AmariliSoft**
977 East Stanley Blvd.
Suite 226
Livermore, California 94550

QuickGIF and QuickGIF Plus are licensed software to the end user. The user agrees to use this software on a single CPU or machine at any given time. Volume licenses are available on request. You may make additional copies only for archival purposes. If you decide to keep using QuickGIF after a trial period, you should pay the registration fee of US $30 and agree to the terms of this license. Quick GIF Plus, on the other hand, is licensed directly to registered users and cannot be distributed, unless as approved by AmariliSoft.

The software is copyrighted material and remains the property of AmariliSoft. It may not be modified, reverse engineered, disassembled, or derivative works created from it. Any special use of the software, not contemplated in this documentation, requires a special license from AmariliSoft.

This license is effective until terminated. You may terminate the license at any time by destroying all copies of the software and documentation in your possession. AmariliSoft may terminate the license at any time by providing you with notification. The license is also terminated if you fail to comply with any of its terms.

Yes, I want Version 1.6 of PCXtc!

I understand the upgrade adds the ability to save PCX files as MacPaint files, and print them directly.

Name ____

Address ____

City ____

State/Province ______________________________________

Zip/Postal Code _____________________________________

Cost: $12.00 (or send $10, plus a blank disk.)

With a self-addressed, stamped envelope.

Send to:

John Papiewski
PO Box 7444
Elgin, IL 60121-7444

Index

A

B

D

E

F

I

L

M

N

O

Q

R

S

T

U

V

CUT TO OPEN

The Scanner Toolkit, Macintosh® Edition
by John Papiewski (PCXtc), David R. Fry (GrayView), Frankie Sierra (QuickGIF), and Electronic Arts (Studio/1)
for use with
The Complete Scanner Toolkit, Macintosh® Edition
by David D. Busch

Requires Macintosh® computer. Version 1.1a. To start the software see Chapter 18, Using the Software Diskette.
BOOK 15-3448-01 ISBN 1-55623-480-5

THE COMPLETE SCANNER TOOLKIT
Macintosh® Edition

DAVID D. BUSCH

If you're a scanner user who is hungry for new ideas and clever techniques, *The Complete Scanner Toolkit* offers the tools and tips you're looking for. Author David D. Busch provides hands-on advice for using scanners in real-life situations, including desktop publishing, advertising, and information management. You'll find a broad range of useful information, including:

- Advice for getting the most from your memory and other hardware.
- Information the vendor won't tell you about installing your scanner.
- How to compile your own clip art library.
- Advice on color separations.
- And much more!

The Complete Scanner Toolkit doesn't just concentrate on the theory and technology behind scanners. It's a practical, hands-on guide for specific applications you can use today and includes the tools you need to make your job easier. Busch gives you a disk of valuable special utilities that includes an animation program, Studio/1, a greyscale program, GreyView, a Gif viewing program, QuickGif, and a conversion program, PCXtc.

About the author . . .

The author of over 30 computer books, including the companion volume *The Complete Scanner Handbook for Desktop Publishing*, and more than 2,000 magazine articles, David D. Busch is the first person to win a "Best Book Award" twice from the Computer Press Association. Before Busch began writing best-selling DOS-oriented user guides, he was a monthly columnist and a contributing or associate editor for six different computer magazines.

ISBN 1-55623-480-5

15-3448-01