Computer Enhancement of Digitized Images Using an IBM PC or Compatible Computer
A Practical Guide

By Terry D. Moore

Table of Contents

Authors Note1-1

Purpose2-1

Introduction3-1

Image Editing4-1

Image Data5-1
- **Data Types**5-1
- **File Types**5-6
- **File Conversion**5-7

Hardware6-1
- **Image Capture**6-1
- **Computer**6-6
- **Input**6-8
- **Monitor and Video Controler**6-8
- **Memory**6-12
- **Storage**6-12
- **Display**6-15
- **Hard Copy**6-16
- **Modems**6-18

Procedures7-1
- **Things to try**7-4
- **Things to Think About**7-7

Software8-1
- **General Tools**8-1
- **Format Conversion**8-3
- **Text**8-3
- **Filters**8-3
- **Color Separation**8-4
- **Compression**8-4
- **Inexpensive Programs**8-6

Getting Started9-1
- **Image Editing Programs**9-1
- **Compression**9-2
- **Miscellaneous**9-2
- **Systems**9-2

The Inside Story10-1
Image Filters10-1
Halftone or Dithering10-11
Charge Coupled Devices10-14
Color Separation10-16
Macintosh Files10-16
Compression10-17

The Future11-1

Technical Note12-1

Appendix AA-1

Appendix BB-1

Appendix CC-1

Appendix DD-1

Appendix EG-1

ReferencesR-1

IndexIndex-1

Figures

Figure 1 Equalization3-2
Figure 2 Histogram4-4
Figure 3 RGB Color Model5-3
Figure 4 RGB Color Space5-6
Figure 5 Normal Edge6-2
Figure 6 Anti-aliased Edge6-4
Figure 7 Two Step Anti-aliasing6-6
Figure 8 CPU Speed #16-7
Figure 9 CPU Speed #26-8
Figure 10 Halftone 16-12
Figure 11 Halftone 26-16
Figure 12 Color Domains6-20
Figure 13 Gamma Curves7-11

Tables

Video Standards6-9
Image Size6-13
Storage Media6-13
Relation of DPI to Lines6-17
Modem Protocols6-18
Image Editing Systems9-3

Authors Note

This paper is a general guideline for those people just starting, or expanding their skills in digital image editing. It is based on a desire to enhance petroglyph and pictograph images. Some information herein is based on practical experience, some on articles read in magazines and books. Yes, I know, the references section is not detailed. Too much to track and type! Soon, I hope to include feedback from readers of this paper. I cannot promise to respond to all communications, as I always seem to be very busy, but I appreciate all input\information\ideas you can give me.

Not all of the methods, software, or specific equipment described, have been tried by yours truly. They may not work as advertised, or they may not be useful to your particular situation. Use this paper as a starting point, a guide, an instrument to pique your curiosity, but not as a bible. After all, wouldn't you get bored if I told you everything?

This paper is available on floppy disk in WordPerfect 5.1 format for those who wish to print on their own (better) printer.

Much thanks to Gerda Galiob, Wilson Turner, Alex Patterson, Dale Harvey, Mar-Sue Ratzke, Pat Logan, Ed Scott, my mother, and all the others who have given me valuable input for this paper.

Version 2.6 August, 1993

Terry D. Moore
10361 Cliota
Whittier CA 90601-1711
(310) 692-1523

You can also contact me, via electronic mail, at:
CompuServe 70711,1014
Internet tmoore@tanstaafl.win.net
Graphics Confrences on RIME and Wildnet networks.

Purpose

The purpose of this paper is to inform the reader of the possibilities currently existing for the editing of digitized images using an IBM PC or compatible computer system. Much of the information is applicable to Apple Macintosh, or other micro-computers or workstations. This paper will address possible enhancements and modifications to digitized images, with a focus on enhancing the image for better quality of image displayed. Image capture, editing, output, as well as software and hardware requirements will be addressed. The focus is on enhancing petroglyph and pictograph images, but much of the information is applicable to other types of images, most notably astronomical images. Most of the information herein applies to color or gray-scale images, with little information on true B+W images. As the amount of information about this subject is staggering, this paper will probably raise as many question as it answers. I have tried to group information into subjects, but all the subjects tend to overlap, so please read all of this paper to gain maximum information about each of the subjects covered. I will revise this paper as new data becomes available, and as I gain more experience with image editing.

Are other people doing this? Are there programs out there to do the things we want to do? Are there user groups involved in true image enhancement (not just graphics and clip art)? Is there a BBS for image editing? Let me know if you have a good lead.

Here is what I have found:

1. There are many BBSs' for "graphics". The term "graphics" is used to describe anything visual, from graphs to clipart to adult material. These BBSs' are often a good place to find the latest version of shareware programs dealing with graphics.
2. There are three major networks carrying confrences worth monitoring. They are the graphics conferences on the RIME, UN'Net, and Wildnet networks. They are usualy accessable through a "Mail Door" on a BBS. An Off Line Mail Reader is the way to best utlize these systems. These confrences are intresting to follow, and can be very helpful when you have a question! The Desk Top Publishing, Ray Traceing, and Virtual Reaility confrences may also be of intrest.
3. Of the three, the RIME net has the most activity in the area of digital imaging and graphics.
4. The Graphsup and Photoforum (Digital Imaging section) on Compuserve are a good bet for more information, and for programs. You can get lost in here. You may want to try WinCIS, or other Copuserve access manager, to handle this place.
5. Just droped into the Internet via WinNET. Will let you know what I find. This is one BIG chunk of cyberspace!

Introduction

Using computers to manipulate digitized image data was for many years only available to those people who had access to a mainframe or supper mini-computer, and the ability to write their own programs. Only a large organization, or a rich individual, could afford the hardware needed for image processing. The software was often specialized for a specific task. Digital image manipulation was only found in large corporations striving toward a particular goal, or in colleges and universities doing research. The personal computer has come of age, bringing the hardware necessary for gathering, processing, and out-putting digital image data to a level of availability unheard of ten years ago. The equipment necessary for a "digital darkroom" is not only available to many companies and schools, by also to many households. More importantly, much of the necessary image handling software is available off-the-shelf. If you do not have the necessary equipment available at home or work, try your local college. (Note: the applicable class may be listed under art, photography, computer science, or other heading. Look for key words like Scan, Scanner, Draw, Graphics, Digitize, Paint, Edit, and Image used in conjunction with Computers. With image editing being used in the medical field, it may be a good idea to check the medical/nursing section too).

I must note that at this time the computer tools available, as amazing as they are, still need much refinement. Unfortunately, many of programs I have dealt with are more involved in processing an image for publication or to create computer art (no problem, I like art, but that is not our aim here). Many are strong on paint, weak on image enhancement. There is still much room for improvement in the area of image enhancement equipment and software. Don't be afraid to ask for what you want. If your software, hardware, or the manual will not do what you want it to do, or give you usable information, call, or write, tech support or marketing, and ask for it (politely, but with determination). Quote this paper, and tell them Terry sent you (boy, that should scare them!). With the vast improvement in software and hardware the last few years (and a little push from us) I have faith that more useful tools, and information, will be available soon.

In this paper I will discuss DOS and Windows programs. Please remember that OS\2 will run both DOS and Windows programs, and is a good choice for an operating system.

Image Editing

Manipulation of recorded images has been around almost since the first photographs were produced. Lightening, darkening, contrast change, double exposure, time exposure, and paint brush or air brush touch up are just a few of the tools. In recording of rock art, our job is not to change an image just to make it more interesting, prettier, or more artistically pleasing. Rather, we want to capture a visual image and convey that image in a manner that, while perhaps enhanced so that we may obtain the maximum information from the image, will not distort its original meaning. The images we work with are often difficult to see in detail due to the passage of time and weathering by the elements. Often, they have been photographed under poor lighting conditions, and sometimes with less than optimal photographic or recording technique. Hand drawings and sketching may miss detail, or may contain information biased by the artists interpretation of the object. Cultural bias also affects the way we see things. The computer is a resource we can use to overcome some of these problems.

First on our list of possible manipulations to an image are corrections to the brightness and contrast of the image. Most programs will have controls like those shown in Figure 1, page . Changing the brightness may help to bring out detail in the image. Contrast adjustment, and often times deliberate over-adjustment, can be a powerful tool. This is especially true of a petroglyph on light colored rock. Highly exaggerated contrast may bring out many details not seen in the original image, causing the image to stand out in stark detail.

The Histogram is a helpful tool in deciding the amount of contrast adjustment. See Figure 2, page . The histogram is a count of the number of pixels (dots) of each color, or gray-scale, level. By adjusting the histogram we can adjust the difference between the lightest and darkest parts of the image. The human eye is more sensitive to changes in light (achromatic) than to changes in color (chromatic). The histogram can help in adjusting contrast without adding visual "noise" or removing detail. Please see Appendix A for examples of contrast, and other, manipulation. Appendix E shows various Histogram\Gamma curves. Please note that many of my examples are gray-scale, but the same processes can be applied to color images.

Perception can be defined as the differentiation of figures from their background. Cultural bias can define how we "see", "understand", or "recognize", an image. Most of us tend to look at a bright or colorful spot, and not at the surrounding shadows. We are used to reading dark type on light paper. Often times a petroglyph will be a lighter image on a dark background. Doing a simple reversal to create a negative image may cause us to notice detail we did not "see" before, or to be more "comfortable" with the image. Looking at both a color image and a gray-scale image side by side may help us in seeing form, or detail. The most simple image editing programs should do these manipulations quickly and easily.

A Magnifying Glass (Zoom Tool) allows us to magnify image as seen on our monitor. In some programs, this can be as much as 3200% (32:1). The resolution of your equipment will limit the usefulness of high magnification but everybody should easily benefit up to 500%, probably more.

Many of the changes described above are ones that, while perhaps easier to do on a computer, could be done in a darkroom with a little time and effort, yielding good results in hard copy form (more on hard copy later). Let us turn to some computer specific functions, things you can't do in the darkroom. Some programs have a Sharpen Function. This feature is somewhat like the contrast adjustment in that it makes dark colors darker and light colors lighter, but it can be set for a threshold (contrast between individual pixels). The Trace Edges Function will try to find areas of high contrast transition and trace the edge of these areas. There is some times a Magic Wand (select tool) that allows you to select a single item from an image (by finding the edges) using a controllable set of guidelines. A Combine (Merge) Function will allow us to combine several images together. Not only additive, as in a normal photographic overlay, but also subtracted, multiplied, divided, and with other methods that offer a whole new realm of image processing. A process called Registration allows us to properly align combined images if needed. We might want to capture many images of the same petroglyph, each with different lighting, and then combining these images. This procedure could be used to minimize lighting effects or shadows.

Being able to manipulate individual colors can be used to enhance color pictographs. If an image of a pictograph contains red, we can amplify the red only, leaving the other colors alone. Or vice versa. (In a red, green, blue image, any color is made up of all three colors. Maybe only 0% of any one. So this image manipulation may only apply to image that has been separated into its RGB components). Many programs allow the creation of custom filters which opens many avenues to explore, and techniques to be developed.

Adding false color to a gray-scale, or converting a color image to gray, may be useful in increasing our perception of the image. This can be very useful to astronomical photographers.

Most programs that manipulate images, especially paint programs, contain tools to be used by "hand" (user manipulated). These tools are often called painting or drawing tools, and usually reside in the "tool box" section of the program. Tools may include a select tool, paint brush, eraser, air brush (spray can), cloning tool, cut, copy, paste, smudge, smooth, etc. Using these tools we can edit or enhance an image by hand, editing down to the level of an individual pixel (dot). Caution is necessary, as artistic bias might distort the final output. However, hand editing is especially good for prepping an image for output to a T-shirt, coffee cup, or publication where extreme realism is not important, only the outline and major details are needed.

Image Data

As you start editing images you will soon find that the size of the image, both as a data file and a physical image, is highly variable. File formats may not transfer between platforms. Some programs may use one type of data, other programs may use something else. A little background information is in order before we can understand this subject.

Data Types

Some graphics programs, designed to generate computer images, such as CAD (computer aided design), draw, illustration, and presentation programs, use a method of data handling called Vector, or Object-Oriented, graphics. To draw a circle the program designates the center of the circle, the width of the line, the distance of the line from the center of the circle, and the uses a mathematical formula to draw the circle. The circle exists as a set of instructions on how to draw the circle. A single line can be defined by its width and its two end points. A square can be defined by the line width and the top right and bottom left corners. Vectors (mathematical formulas) are used to describe each item in the drawing. This makes it much easier to move from one device (monitor, printer, plotter, etc.) because each object in the drawing can be re-sized for the target device by using mathematics. Nothing is lost, or distorted, unless we overrun the capabilities of the device. Some of the figures in this paper were created using a vector drawing program. Unfortunately, the real world is too complex for vector information.

We will be using what we call Bitmapped (Raster) images. This method is better suited to the complexity of real world images. A good example is scanning a photograph using a flat bed scanner. The scanner starts at the top left hand corner and reads the information (reflected light) the first sensor dot (photosite) receives. It then moves or switches to the next sensor to the right (mechanically or electronically) and records the information it receives. This continues to the right edge of our photograph. The scanner then moves the sensor array down a step (mechanically or electronically) and scans the next line, until we reach the bottom of our photograph. In this way, the scanner reads the data in much the same way we read a page of text. The scanner may do this scan operation once, or three times (for a color) image, depending on the design. Each dot of information can be 1 bit of data for black and white (line art) to 24 bits of data (8 of red, 8 of green, and 8 of blue) for True Color. The same is true of a video camera, where light reflected from the object is focused on an array of sensors. The information is then read off the sensor, spot (photosite) by spot. The result is a map of the information received. (A purest might contend that only a 1 bit scan (B+W) is a true Bitmap, but the term is used for multi bit images as well).

Most paint, scan, and screen capture programs (these store an image from your monitor to memory, the Clipboard, or a disk file) use this bitmap method.

Since each dot contains information directly relating to the original, you are now "locked" to the image. If the picture you scanned created a file with more dots then your monitor can display, you will only see as much picture as the size your monitor will allow. The same is true for printing. If you have a picture with 2000 dots on each line, and your printer can only print 1500 dots per line, you lose 500 dots of the picture. Image editing, and many other programs, can re-size (resample, rescale) the image to fit. However, they must add "made-up-data" to expand the image in size, or subtract data to shrink the image in size. Is the data lost really important? Is data added really a detriment to the true image? This judgement must be made on a case by case basis, depending on what information the image should convey. Different programs may add or subtract dots in a different manner, giving different results.

All of this worry about file size when digitizing data can be a real exercise in frustration. The good news is that many times you may not need to worry about it. Later in this paper I will stress more and more the need for high quality professional work, with all enhancement to an image done deliberately. Each change made with a specific, and understood, method and purpose. When starting out, just make things work. Slowly learn to do better quality work with each experience, honing your skills until the only changes you make to an image are for the good. Don't let the complexity of the whole process scare you. Getting started, and doing interesting and valuable work, is not hard at all! The only thing to remember when getting started is never do anything to the original (archive) data that you can't undo! Once you have an image, make a copy of it for storage. This way you can always go back to the start.

Color Gamut

If working with a color picture, the color data can be in several different formats. Most computer systems use RGB color. This is red, green, and blue. This is an additive system. A computer monitor (or color TV) is a good example. The three colors are added to make other colors. All three equal makes a shade of gray. All colors full on makes white. All colors off makes black.

RGB color is different from subtracted color. In mixing paint, the three primary colors are yellow, red, and blue (sort of). In printing CMYK (cyan, magenta, yellow, black) colors are used. This is called subtractive (reflective) color. White light - red light = cyan (blue+green). The K stand for black and is used as a fourth channel to produce better print quality. Black ink is inexpensive, and easily used to adjust brightness.

The third, and much rarer, color scheme is HSV. This form is hue, saturation, and value. Hue is the color without black or white (color). Saturation is the amount of gray (vivid or quiet). Value is luminosity (bright or dark). This is a good system to use as it is more intuitive, much easier to understand, than RGB.

Figure 3 shows a model of the RGB color scheme. Figure 4 shows the RGB color space. Point A is black, and as you approach point B, you approach white.

Figure 12 shows the relationship between the Commission Internationale L'Enclairage (CIE) international standard for specifying color. This defines the color the human eye can see. The monitor space is a standard color monitor, and the printer space is a four-color offset lithography system.

Color Depth

The number of spots (dots) per inch our sensor is capable of receiving sets the resolution (sharpness or detail) of the image captured. The image on your systems monitor is displayed by a reverse of this method, projecting information dot by dot, instead of capturing the image dot by dot.

The number of levels captured by each photosite sets the depth of gray or color information per dot (two shades, 64 shades, 256 shades, etc). This is known as color depth, sampling frequency, or bits per pixel.

In the computer binary data is used to control the number of colors in a image. One method of doing this is to assign a specific number of bits to each of the electron guns in the monitors CRT (cathode ray tube). On a true black and white system we need only one bit, on or off. Some monochrome systems can use two bits, for a total of four shades. Some monitors support color by having three guns, one red, one green , and one blue. The number of colors your system can display is controlled by the monitor and video controller.

The number of colors available is controlled by the type of file and the color depth the image is stored at.

Grayscale is usually strait forward. 1 bit is black and white, 2 bits is 4 shades of gray. This continues in binary fashion to 8 bits for 256 shades of gray, the current limit for most equipment. Some equipment will go beyond this to 12 bits.

Color is much the same, but each pixel has three numbers, one each for red, green, and blue. A 24 bit color image is 8 bits of red, 8 of green, and 8 of blue. A 15 bit image is 5 bits each of red, green, and blue.

Indexed color, is a bit harder to understand. Perhaps the explanation I wrote for a question I answered on a network will explain:

-> You can convert a GIF to Targa format, but what you
-> get is a GIF image stored in Targa format. The converted
-> file will still have 256 or less colors.

KP> Indeed, but it *will* be a 24-bit Targa image, won't it?
KP> I agree that it could have looked better if the source format
KP> wasn't limited to 256 colors, but that's not the issue right   KP> now.

The issue is whether or not a 24-bit Targa image can be KP> created from a GIF (so long as the GIF has 24-bits per color KP> map entry to begin with).

You can make a Targa file from a 16 color PCX, or a b&w PIC, a 15 bit high-color image, or any other format you want. All you need is the software.

If the software does its job, you will end up with a 24 bit Targa that can be used like any other Targa file. It *will* be a 24 bit Targa file. It will be in Targa format, will have a Targa header, it will use 24 bits per pixel. Will it be usable? Yes! Will it be a waste of time and space? Yes!

I can open a new file in a paint program and fill the image with black. Every pixel black. I can save that as a Targa 24 bit image. Each pixel would be defined as 000 red, 000 blue, 000 green. If I fill the image with white, the values would be 255 red, 255 blue, 255 green. Three BYTES to do what I can do with one BIT. But it *is* a 24 bit Targa file.

If I put a picture containing 51,226 individual colors on a scanner, scan, and save in 24 bit Targa format, I get an image with 51,266 colors. This is possible because the 24 bit allows me to store 256 reds times 256 green times 256 blue for 16.7 million combinations for each pixel. More than enough for my 51,226 color needs.

If I do the same, but chouse to save in GIF (an 8 bit indexed format), things are vastly different. The palette (color look up table, color map) defines a color by 8 bits red,8 bits blue, and 8 bits green. 24 bits, yes? But there are *only* 256 slots in the palette. You can only define 256 colors. Yes, they are 24 bit colors, but from the original 51,226 you can only select 256 to use. So, saving in GIF format is a form of color reduction. You can use a "standard palette" in which the colors are predefined. Any color between "a" to "b" becomes color #1, any color between "b" and "c" becomes color #2, and so on. If I define my "standard" palette as 256 shades of brown, I get an image that looks like an old photograph.

You can use an optimized palette. The software then looks at the image, analyses the color content, and the selects the best 256 colors to use. For example, a program might will take 4 colors in a current image

245 red 000 green 147 blue
245 red 000 green 134 blue
240 red 010 green 149 blue
254 red 011 green 141 blue

and make them ALL color # 5 which is defined in the color table as 245 red 005 green and 142 blue in my palette. Our color reduced image now uses color #5 any where that any on the original 4 colors was used in the original image.One byte to define a pixel color (plus the overhead of the map) is a reduction in file size over three bytes per pixel. The trade off is the loss of *simultaneous* color possibilities.

A optimized paulette might look something like this:

color	red	green	blue
0	000	127	251
1	000	127	011
2	152	015	132
.			
.			
.			
255	162	162	000

Once we have done this color reduction there is no way to recover the data. How can a program know that color #5 in our make believe palette is being used to simulate 4 different colors, what those colors are, and where they belong? It can't. The original data is gone forever.

The palette for a different picture may (will) be completely different.

The scanned image can be converted from a GIF into a 24 bit Targa file. Will it be the same as the one original scanned as a Targa 24? NO! The original Targa will have 51,226 unique colors, the converted image will have only 256 unique colors. Both *are* 24 bit Targa images. They are both the same file size. The converted image is a waste of space, using three bytes to do what we can do with one byte. (Note how I worry about file size. My disk get smaller every day!)

This applies to ALL indexed (paletted, mapped) file formats, not just GIF.

This also applies to any programs that are hardware dependent, and the hardware is palletted. That is to say, if you display a 24 bit color image on a 256 color VGA monitor, and use a screen capture program, you will get 256 colors only.

File Types

Images can be stored as computer files in many different formats. TIF (TIFF), PCX, BMP, and TGA (Targa) are the main bitmapped file types (file formats) used for full color (24 bit) images. The GIF (very popular), WPG, and PIC are popular 256 color (or gray scale) formats, with GIF being stable and widely used. GIF is 256 colors selected from a palette of 16.7 million colors. These are only a few of the many formats you may run into.

The TIFF format, though popular, is plagued with compatibility problems. It was designed to have many options, and so ended up with many variations. Some of these variations may not be supported by your program, and some do not conform to the TIFF specification. These problems are slowly disappearing as TIFF usage becomes more wide spread. The TIF (TIFF on the Macintosh) format is almost the same on PC and Macintosh except for the header. Using a simple program to fix the header may make it easy to transfer images from Macintosh to the PC and back. But, due to the complexity of the TIF specification, this is not guaranteed.

One of the benefits if the TIFF format is an anility to store data in compressed or un-compressed format. The data compression, up to this time, has been losless and therefor useful. TIFF 6.0, also known as TIFF/JPEG, is the TIFF file format with the JPEG lossy compression. If you use the TIFF file format with compression, make sure you know if it is losless or lossy.

TIF, despite its problems, is one of the major players in this game and must be dealt with. I hope the current trend toward workable, interchangeable, TIFF files continues.

The TGA (Targa) format was designed in support of a series of video controllers, and has somewhat limited support (much as the Hercules Video Boards have somewhat limited support).

The PCX and BMP formats are somewhat more stable. These are the ones to use if presented with a choice.

Other bitmapped file types include BBM, CE, CUT, DIB, LBM, MSP, RIX, RLE, VMG, and many others. File conversion programs, many of them very inexpensive shareware programs, are available if your editing package doesn't handle a needed format. You must check to see if they do a correct, or a fast and dirty, conversion. Some of these programs may not give acceptable results.

If you need a vector format (HPGL, GEM, CDR, DXF, EPS, Windows Metafile, and others) there are conversion programs available. Or vice vera. Conversion of bitmapped images to vector format will result in a less than desirable image. Vector images are suitable (even desirable) for design, drafting, industrial graphics, line art, but not for the complex, real world images we work with.

File Conversion

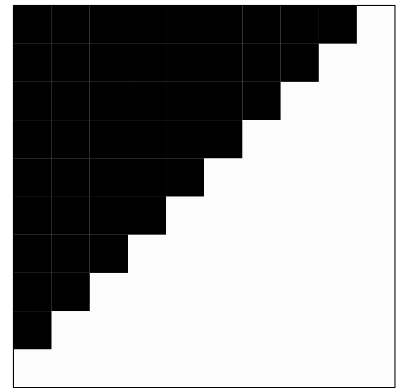
The same problems we have in resizing an image occur when we convert color to gray-scale, such as 8 bit indexed to 24 bit color, or other conversion. The program must decide what to keep, what to throw away, what to change, and what to create itself. In gray-scale to color conversion, the image can be very good, but remember that it is not "real", and may distort an image in a manner we may not want. Color to gray-scale is usually very good, and useful.

Problems can exist when recording, or converting to/from, one file format to another. Converting file format may be necessary because most programs support only a few of the many file types. Some formats support an optimized palette. The optimized palette format can support the color, or gray-scale, data exactly (or very close to) the supplied information. Conversion programs that can do optimal palette format may use algorithms such as Heckbert's Median Cut. Other file formats, such as TIFF 16 shade gray-scale, and some conversion programs, use pre-set shades and therefore will convey only semi-accurate information in conversion. For example, a gray-scale image in 16 color PCX format converted to 16 shade gray-scale TIFF will not contain the same shades of gray. The program will convert the real gray-scale data from the PCX file to the preset shades the TIFF file uses. To avoid this problem you could convert to a 256 color TIFF file which allows the use of the exact shades. Some file formats may contain both bitmapped and raster format information in one file and any conversion or editing can cause loss of data. Different formats may have the same extension. For example Lotus 1-2-3, Dr. Halo, and several other programs use the .PIC extension.

See Appendix D for more information on file formats.

Hardware

I will attempt to give some hard numbers to work with. Giving solid information is difficult because the numbers will vary depending on the type,



Please note that during this discussion I will refer to 1, 8, 24, and 32 bit images. I am referring to the file type, or the function of the hardware, and not the physical type of slot that the board fits into. IBM and compatible systems can contain 8 bit ISA and\or 16 bit ISA slots. Some may have 32 bit EISA, MCA (microchannel), or proprietary card slots. Often, systems contain a

The new systems may have a VL or PCI (local) bus. This is a high speed bus, and a video card on this bus will be <u>much</u> faster than one on a standard ISA bus. If you geta new system, get one with the new bus, and a video card to match.

Check out the hardware package before you buy. Will the disk be big enough? Does the 24 bit video card come with a driver for DOS, will it work with the current driver, or does it use pass-through? How about a driver for Windows if needed? Same thing applies to printers and other devices. Do you have a slot open to plug the board into? Is it ISA 8 or 16 bit? Is it EISA or perhaps MCA (microchannel)? Look closely before you buy.

Image Capture

The first thing we need to do is to capture the image. There are several ways to do this, and each method has strong and week points. The most obvious is to take a picture using a standard camera and digitize the picture. The second is to use a video camera. The third is an electronic still camera.

At this time there are several types of electronic still cameras.

Analog electronic cameras use a CCD (charged coupled device) to capture the image. See the chapter The Inside Story for more on CCD's. The image is then stored in analog format, usually on a 2" disk. Some have an external playback device, some have playback built into the camera. Output is usualy a TV compatible video signal so you will need a frame grabber to get the image into your computer, and are limited to the resolution of TV.

Digital electronic cameras use CCD's, but record the image in digital format. Storage is by internal or external disk, ram memory, or IC memory card.

At this time the commercial electronic still cameras are not a viable alternative. Not only too expensive for what you get, but many have a resolution is too low for all but the most inaccurate work. Not worth the price at this time ($700 to $20,000). In a few years this should change. I eagerly await the day when we will have reasonably priced digital electronic cameras designed to interface easily with computer systems.

One major plus for images captured directly to the system, provided they are from a stable photographic setup, is that the images can be combined (merged) without having to align (register) the images. Several images (frames) captured directly in to the system will be the same size and start at the same location. Images captured to film, converted to prints and the scanned will not be aligned relative to each other.

Input from a video camera or a video tape player is now available via the use of a frame grabber (video capture) board. I do not recommend input from a video tape as they tend to have too much noise (dropout) causing poor image quality. To some extent, this can be overcome by capturing several duplicate images and averaging them together. This, however is time consuming and requires storage space for many images.

Direct input from a video camera (using an image tube), graphics camera, or video camera using a CCD into a frame capture (grabber) board, is a better solution as far as image quality is concerned. Bypassing the noisy recording on the video tape will improve the picture quality. Beware of video equipment for use with TV. If you have the equipment on hand, use it. You may, however, find that the resolution is limiting. The standard television is not a high resolution device. Current TV standards (circa 1941) are 525 horizontal lines interlaced (two scans to make one image). A typical TV system is approximately 480 horizontal scan lines by 380 pixels per line.

Video bandwidth is scant 7Mhz (million cycles per second) for TV. Bandwidth may be as high as 110Mhz on a high quality computer monitor. TV looks marginally good because of its fairly wide color capability, and the fact that it is mormaly a moving image.

The recent innovation, Super VHS, is an attempt to improve TV by increasing the brightness (luminance) bandwidth, but does nothing for the resolution. If you use color you will see improvement using SVHS, but may not realize any gain in monochrome. Real HDTV would be here but for....... well, never mind. Let us not talk about politics/money, and getting stuck with second best.

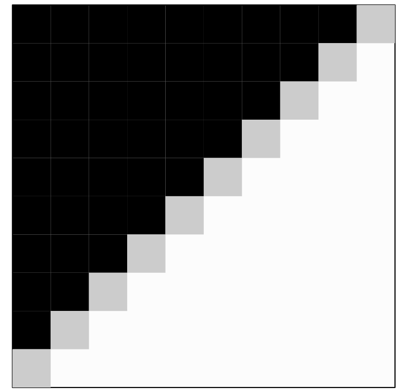
When HDTV arives it will probably be a 16 to 9 aspect ratio at 1920 by 1035 pixels.

Sony has a 20 in square, 2000 x 2000 pixel monitor. Sharp has demonstrated a 1280 x 1024 color flat panel LCD display.

Even though the standard for TV is 525 lines this number will vary on a unit to unit basis. Some cameras made for industrial or scientific use may be 700 lines or better. An inexpensive camcorder may be as low as 200 lines.

See the section on Film\TV resolution for more information regarding TV and video camera resolution.

If you do decide to go with a frame grabber type video system you may wish to consider using a closed-circuit or surveillance camera as your input device. They are small and inexpensive when compared to a top of the



line camcorder.

Note: I stress using the best. If an inexpensive camcorder is what you have, use it! You may be surprised at how well it works. Get started now, and upgrade as needed.

Our goal in recording and enhancement is to start with the best possible picture quality. There are now hi-resolution video cameras available. Some of these cameras are made for use with microscopes or telescopes, and have much better resolution and bandwidth then a standard video camera.

Since you cannot install expansion boards, such as a frame grabber, in a laptop or notebook computer, this limits direct video capture to a full sized computer system. Computer, camera, power supply, lights, and other support equipment will make a bulky package.

Slides and film negatives can be digitized using a film scanner. Currently, the price of a 35 mm slide digitizer is $3000 to $10,000 for 24 bit true color. Film scan resolution is normally given in lines. 4000 is 4096 x 2732 pixels (dots), and is about the limit of high quality film. See the section The Inside Story for more information about film resolution. Input directly from slides or negatives will bypass distortion that can be added during the conversion to print and the scanning of that print.

Before running off and having your 35mm slides scanned at 4000 it time to think about file size again. At 4000 and 24 bits, the resulting file will be 32Mb! Most film scanners will do 2000 (2048 x 1366) and 1000 (1024 x 683). A 2000 line scan will result in a 8Mb file and a 1000 line scan will yield a 2MB file. 256 color or gray-scale scans will reduce the file sizes to 10.6Mb, 2.66Mb, and .66Mb. Can you scan only a section of the image at high resolution? A good question, for which I do not have a good answer. There are service companies that will take your slides or prints and return the digitized image for $10 to $20 per image, but it may cost more for custom editing.

Kodak is now putting photos on compact disk. You drop off your exposed film, and later pickup what Kodak calls a Photo CD. Each CD can hold about 100 photos. Color depth is 24 bit, but they left room for up to 16 bits per pixel if needed! Huffman compression (losless) is used.

Each image is stored in:

Base/16	128 x 192
Base/4	256 x 384
Base	512 x 786
4*Base	1024 x 1536
16*Base	2048 x 3072
64*Base	4096 x 6144

The 64*base is an 18Mb file. This is the reason I suggest big disks and large memory!

Kodak offers a Photo CD player to attach to your TV, or to a frame grabber installed in your computer.

The Photo CD disk can be read (mostly, there are some compatibility problems) by a computer system that has a CD-I or CD-ROM XA drive attached. (CD-ROM XA is a subset of CD-I) The CD player must support multi-session, or all of the images must be transferred to the CD at one time. Cost for Photo CD processing is about $1 per picture. One disk will hold about 100 images.

If you have compatibility questions about Photo CD and the player you have or wish to buy, Kodak has done testing for compatibility. This information is available in the Kodak Forum on Compuserve, or directly from Kodak.

I expect most image editing software companies will support Kodak Photo CD images soon. Plug in modules for the major image editing packages are already out. Kodak has its own software, Photo Access and PhotoEdge (either one is recommended).

Support for professional film is also available. 120, 70mm, and 4 x 5 are supported with the Kodak Pro Photo CD.

Expected life of a Photo CD is 100 years or better given standard care.

IN MUCH OF THIS PAPER I HAVE USED SCANNING AS THE BEST INPUT AT THIS TIME. I NOW BELIEVE THAT THE PHOTO CD IS BY FAR THE BEST CHOICE FOR MOST PEOPLE.

A scanner can be used to input data from a standard print, magazine, or other "paper pictures". These machines are much like a photocopier in that they scan an image but, instead of giving you a hard copy, they transmit a digital image to your computer. There are service companies that can do this for you at a reasonable charge.

Scanning, print or slide, also makes it possible to edit those boxes and boxes of pictures you already have, rather that taking a new set of images.

Scanners come in several forms. Hand held and fixed (desk top) are the two most common. The hand held scanners can be used for some applications, but have their limitations. Since the scanner is hand held, changes in scan speed, jitter, and skew can introduce errors. Most hand scanners are only 3 or 4 inches wide, although programs, often supplied with the device, can "stitch" several scans together to make a wider image if needed. These scanners are, however, reasonable in price at $100 to $400.

Fixed scanners are mostly of the flat bed type. The picture to be scanned is laid face down on a glass plate and a motor moves the optical scan assembly. These scanners have better accuracy than the hand held, and most will accept 11" x 14" documents. The price for a full color, 24 bit, scanner is under $2000. This normally includes an interface card and a software package to control the scanner. This software package just may be one of the several hundred dollar image editing programs you were thinking of buying. The price difference between color scanner and a gray-scale scanner is $100 to $250 so there is little reason not to go color. The color scanner will do B+W and gray-scale, as well as color. If you have a gray-scale scanner, hand or flatbed, you can still do color by using red, green, and blue filters and then combining the images.

A good flat bed scanner should be capable of B+W, 256 gray-scale (8 bits), 256 color (8 bit indexed), and 16.7 million color (24 bit). The 16.7 million color gamut covered by most monitors is still less than the human eye can see. This is true of RGB, CMYK, and Pantone 747 color gamuts. Although this is less than perfect color it is a good compromise for file size, equipment needs, and quality.

Some high-end equipment can now do over a billion colors by using 30 bits (10 bit each of red, green, and blue) or 36 bits (12 bu\its each).

The scanner resolution should be at least 300 x 300dpi (dots per inch) for quality work. 400 x 400dpi is even better. It should also be capable of scanning at lower dpi settings. 75dpi is a good number for the low end setting. Caution, the advertised resolution may not be the actual hardware resolution! Many scanners use a process called interpolation to "double" the resolution. This can enhance the perceived quality of the image, especially if you will be digitally enlarging the image. Remember that interpolation is adding false information to our image so we will not want to use this function on the archive version of our image. Interpolation may be good for an image that is to be enhanced. Be sure the scanner you get is at least 300 or 400 dpi (or better) without interpolation (600 to 800 dpi or better with interpolation) if you plan on doing quality work. True 600dpi and better scanners are now hitting the market.

Some scanners may half-step the CCD photosite assembly to give a 600 x 300 dpi resolution.

1200 x 1200 dpi 30 bit scanners are available for $5000 for real high quality work.

Color scanners that make three passes to capture an image may be less accurate than single pass color scanners due to mechanical tolerances of the scanner. The same thing applies to printers.

OCR (optical character recognition) is not a plus if you are planing on image processing only. It is for scanning text. Also, scanning in halftone mode is not usable because of image quality and the inability to edit a halftone scan image. See the information in the section The Inside Story for an explanation of the halftone process.

Computer

Now that your image is digitized and ready to edit, what computer do you need to run your image editing program? That depends on what program you use. *Pick your software first*, and then buy a system to match! There is nothing worse than buying a system that will do many things, but isn't up to doing what you want it to do. Check with the software vendor as to the minimum hardware requirements for a given program. Ask what you need to make it work, and what you need to make it work well.

The minimum system for any *serious* graphics work is a '286. Better yet is a '386 or '486 system. If your software runs under Windows or OS/2, a '386 or better is mandatory. The faster the system processor the better, 20 Mhz is Ok, 33 Mhz and up is the way to go. Depending on the software, a Math Co-processor may provide a big speed improvement during image processing. Make sure the system you chose has a co-processor installed (it is built into a 486DX), or can be upgraded later (less than $200).

The following chart shows processor speed. Please note that these speed ratings apply to the Intel line of processors only. Actual system speed will vary depending on the way the system was designed.

More CPU speed ratings.

See the section entitled Getting Started for more system information.

Input

Most programs will need a mouse, or the equivalent, to work well. If you plan on doing some fancy drawing, tracing, or other hand input you may want to consider a mouse pen, track ball, or digitizer tablet. These are not mandatory and may not be of any help to you depending on your application. You will almost always need a mouse.

Monitor and Video Controler

If you only do gray-scale images, you can use a decent quality monochrome monitor. There are monochrome only monitors that will provide better quality at the same resolution than a color monitor (a single gun vs three guns and the mechanical tolerances thereof). It's hard to justify the price for a super quality monochrome monitor unless your entire system is setup for gray-scale only, and most of your work will be done with the monitor as the major output device. Read on about the resolution of monitors, and other, devices.

Most people will want to go color, as a color monitor will do B+W, gray-scale, and color. For maximum compatibility vs price make sure to get, or be compatible with, Enhanced, Extended, or Super VGA (256 colors-8 bit indexed, 640 x 480 pixels). This works out to about 70dpi on a 14 inch monitor. A SVGA (Super VGA) system at 800 x 600 pixels (16 color) is about 75dpi. Please note that 256 color VGA can handle 256 simultaneous colors (from a palette of 262,144 possible colors) but only 64 gray-scale tones. This is because the hardware is 8 bits of indexed (mapped, palette) data and only 6 bits per color at the DAC (digital to analog converter). Also note the standard VGA is only 16 colors inch the 640 x 480 mode and only 320 x 200 in the 256 color mode color so ask for 256 color SVGA. (Look for 512Kb to 1Mb of RAM on the video card).

Older EGA is 640 x 350 x 16 color maximum. MDA, CGA, and MGA are even worse. Do not consider these video boards for graphics (image) use.

Video Standards

Name	**Resolution**	**Mode**	**Colors**	**Compatibility**
MDA	720 X 350	Text	1	None

CGA	640 x 200 320 x 200 160 x 200 320 x 200 640 x 200	Text Text Graphics Graphics Graphics	16 16 16 4 2	None None None None None
MGA or HGC See note 1	720 x 350 720 x 348	Text Graphics	1 1	MDA None
EGA	640 x 350 720 x 350 640 x 350 320 x 200 640 x 200 640 x 350	Text Text Graphics Graphics Graphics Graphics	16 4 16 16 16 16	CGA, MDA CGA, MDA CGA, MDA CGA, MDA CGA, MDA CGA, MDA
PGA	640 x 480	Graphics	256	CGA
VGA see note 2	720 x 400 360 x 400 640 x 480 640 x 480 320 x 200	Text Text Graphics Graphics Graphics	16 16 16 2 256	CGA, EGA CGA, EGA CGA, EGA CGA, EGA CGA, EGA
MCGA	320 x 400 640 x 400 640 x 480 320 x 200	Text Text Graphics Graphics	4 2 2 256	CGA, EGA CGA, EGA CGA, EGA CGA, EGA
SVGA see note 2	800 x 600	Graphics	16	VGA, CGA, EGA
8514/A see note 3	1024 x 768 640 x 480 1024 x 768	Graphics Graphics Graphics	16 256 256	VGA pass-through
XGA see note 3	1056 x 400 640 x 480 1024 x 768 640 x 480	Text Graphics Graphics Graphics	16 256 65,536 16	VGA VGA VGA VGA

Note:

1. Hercules Monochrome Graphics Adapter
2. May do 640 x 480 with 256 color. Needs at least 512Kb memory on video board.
3. IBM systems are interlaced. Compatible systems may not be interlaced.

Some SVGA and 8514/A video cards will give you more resolution, up to 1024 x 768, but not more than 256 colors. The optimum system (this week) would have a True Color (24 bit, 16.7 million color) 1024 x 768 pixel non-interlaced video. A 1024 x 768 video system will give you about 2.5 times as many pixels (dots) as a 640 x 480 system. About $2000 for the video card. If you are thinking of upgrading, you will almost certainly need a higher resolution monitor to go with the new video board. 14 inch is Ok, but a 16 inch monitor has approximately 30% more screen area than a 14 inch. A 17 inch monitor has 45% to 60% more screen area than a 14 inch. $1500 to $2500. (The picture may be .5 to 2 inches smaller than the manufactures stated size).

A large monitor, 17 inch or better, will allow you to work with several images at the same time. The 4 images in Appendix A are all in one file. I can only see 2 full images and ½ of the second two on my 14 inch monitor without scrolling. It would be nice to be able to layout six or eight images at a time to edit them without having to scroll back and forth. Comparison of images and enhancements would be much easier.

If you get a new monitor, you may want to get one with adjustment for the RGB mix. This will be useful as more, and better, color calibration schemes become available.

Numbers like 640 x 480 and 1024 x 786 are a 4 to 3 ratio. Displaying at other ratios will distort an image made for a 4 to 3 system. (35mm film is 3 to 2. NTSC TV is 4 to 3)

For the sharpest picture look for a monitor with a dot pitch of .28mm or less. The increase in colors available on the 24 bit system will give a perceived increase in resolution due to the smooth translation from one tone to the next. Look for a "Flat" screen as these monitors will have less distortion than a curved screen.

Get a non-interlaced system. Interlacing takes two passes to "paint" one screen. The odd numbered are painted on the first pass, even numbered are painted on the second pass. This can cause flicker which is hard on your eyes. The video board and monitor should have a vertical frequency (refresh rate) of 60Hz or better for EGA or less systems. 70Hz or better for VGA or better systems. Anything less will certainly be interlaced.

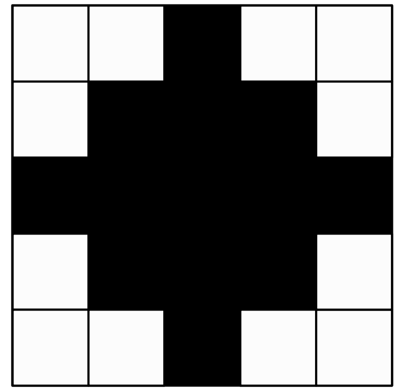
Make sure that any high resolution video cards and monitors you get will also work at lower resolutions, either on-board VGA, or with pass-through from your old VGA card. This will allow compatibility with current system operation. Some video cards and monitors will do 32 bit color and/or 1280 x 1024 pixels. The 32 bit is 24 bits of color and 8 bits (called the alpha channel) for added information, such as transparency. There is limited support for 32 bit or 1280 x 1024 pixel systems at this time.

Do you need 32 bit? It may be useful in art and other aspects of digital imaging, but I have found no reason to go to 32 bit, and the drawback is increased file size.

The new systems may have a VL or PCI (local) bus. This is a high speed bus, and a video card on this bus will be <u>much</u> faster than one on a standard ISA bus. If you geta new system, get one with the new bus, and a video card to match.

Some of the new True Color video cards have an optional data compressor to compress the image file size stored on your disk. See the software section for more information on data compression.

Some video boards are now advertising High Color (not to be confused with True Color). Some are 32,768 colors (15 bit, 5 bits each of red, green, and blue). Others are 65,536 color (16 bit, 5 bits each of red and blue, 6 bits of green). The human eye is more sensitive to changes in green then to red. Currently, there is limited support for these boards, but support is on the way.



Some new boards have options such as anti-aliasing (curve smoothing, edge feathering, continuous-edge). An edge can be defined as a sharp light/dark transition in an image. Figure 3, page shows (magnified) a non-aliased (normal) edge as drawn on a monitor. Figure 4, page , shows an edge with single line anti-aliasing. Figure 5, page , shows an edge with double anti-aliasing. (Please note that what was displayed on my screen when creating this figure with Corel Draw and what was displayed in WordPerfect are both different from the final print. These are the types of problems you will face when working with different formats, programs, systems, etc).

The anti-aliasing function may visually improve the quality of the image see by smoothing the transition from dark to light (or color to color). However, this is a change to the image that you may not want, or have control over. If the video card you are thinking of using has one of these functions, be sure the function can be turned off so you can see the raw image if you want to. This visual change on the monitor will not (should not) affect the image file or final output.

Memory

If you chose a software package that runs under DOS, especially a gray-scale only system, you may be able to run your software in 640KB (kilobyte) or 1Mb (megabyte) of system memory. Chances are you need more memory, or will see speed improvements if you have more memory. Digital pictures are big. Very big. You need memory and disk space to store and manipulate your images. Some programs may use your hard disk as memory if it runs short on RAM (system memory), but this is slow. It can be very slow on older systems with a slow processor speed and interleaved disks. If your software is a Windows application 2Mb of system memory is probably the absolute minimum, 4Mb is looking better. OS/2 users need a least 8Mb. I am almost satisfied with 8Mb on my '386 (DOS and Windows), but I also use memory for a RAM Disk, Print Spooler, and Disk Cache. If you plan on using large 24 bit color images you may want to have 32Mb or more! Memory is $60 to $80 a megabyte.

Storage

A fair sized hard disk is needed. 20Mb may not be enough if you have other software to run, or do large true color (24 bit) images. Lets do some numbers. DOS takes up about 2Mb of disk space, Windows (if needed) takes about 8Mb. (OS/2 may use 30Mb). Your image editing package, you may have several, may take 4Mb to 6Mb each. Throw in a few extras, and we have already used 20 to 30Mb. For picture storage, a B +W image needs 1 bit, 256 color needs 8 bits, and true color needs 24 bits for each dot (pixel). A full page scan at 300dpi and 256 colors will create an 8Mb file (8.5 inch x 300dpi) x (11 inch x 300dpi) x 8 bits per dot ÷ 8 bits to get bytes ÷ 1024 to get Kb ÷ 1024 to get MB = 8.02MB total + file header information). A 24 bit true color scan would have 3 times as many bits per dot yielding a 24Mb file size. Luckily, most scanned images will be photo sized, 3 inch x 5 inch, or less. Any decent scanner control software will allow us to scan only the portion of the image you want, keeping file size to a minimum at the start. You also can crop unwanted image information from the file later using your editing software.

Windows, or other software, may want to use your hard disk for temporary space or virtual memory so you also may need to keep 2Mb to 10Mb free disk space at all times. My 82Mb disk is far too small for the normal software programs I run, plus the images I want to enhance and store. There are software or software/hardware combinations that can compress the image data file so it takes up less space on your disk. See the software section for more information. A fast IDE, SCSI, or EDSI disk with 1:1 interleave is recommended. Older MFM or RLL disks, especially with 2:1, 3:1, or worse, interleave will be too slow for most people. When you look for a new disk, think big and fast.

Image Size

Bits Per Pixel	**1**	**4**	**8**	**15**	**24**
Colors	Mono	16	256	32,768	16,777,216
512 x 512	32K	128K	256K	480K	786K
640 x 480	38K	150K	300K	900K	900K

800 x 600	59K	234K	469K	938K	1,406K
1024 x 786	96K	384K	786K	1,536K	2,304K
1024 x 1024	128K	512K	1,024K	1,920K	3,072K
1280 x 1024	160K	640K	1,280K	2,560K	3,840K

Note 1: Divide file size in kilobytes (K) by 1024 to get size in megabytes (M).

Note 2: File size given does not include header information. Header information is added if the file is stored formatted (GIF, PCX, TIF, BMP, etc).

Note 3: Chart can be used to calculate the amount of system memory needed to hold an image, or necessary memory for a video card.

You also need a device you can use to make a backup of your data. This also may be used for off-line storage of images. Most systems will have at least one floppy disk. New systems should come with high density (bigger than the old Double Density) 1.2Mb 5 ¼ inch or 1.44Mb 3 ½ inch floppy disk systems. People doing true color, large images, or large numbers of images, may want to get a tape, optical disk, or other large storage media.

Storage Media

Drive Type	Capacity	Speed	Price	Media
Hard Disk	10Mb-1.2Gb	Fast	$100-$2000	N.A.
5½ Floppy	360Kb-1.2Mb	Slow	$50-$75	< $2
3¼ Floppy	720Kb-1.44Mb	Slow	$50-$75	< $2
¼ Tape Cart.	80-250Mb*	Very Slow	$500-$900	$35-$75
4mm Tape @	2.5Gb	Very Slow	$2500-$4000	$30-$45
8mm Tape @	5Gb	Very Slow	$7000-$8000	$40
WORM @	1Gb	Medium	$2500-$4000	$100-$200
R/W LASER	600Mb-1Gb	Medium	$3000-$6000	$130-$250
Bernoulli	44Mb to 150Mb	Slow	$1100-$2500	$90-$140
Floptical @	21Mb	Slow	$500-$600	$20-$22

* With built in data compression.

@ So far, these are "niche" products, and may never see wide spread usage.

Tape storage devices (¼ inch cartridges) are $500 to $900 for 80 to 120Mb, and up to 250MB using the supplied (lossless) data compression. Remember that if you already compress your files this second compression may not help. ¼ inch tape is extremely slow and suitable only for backup.

Media is $35 to $75 per tape.

4mm digital audio tape is $2500 to $4000 for 2.5Gb (gigabyte) storage. File access is extremely slow. Media price is $30 to $45.

8mm (video) tapes devices are $7000 to $8000 for 5Gb and, like 4mm tape, are extremely slow. Media price is $40.

WORM (write once, read mostly) Laser Disks are not as fast as a hard disk but are much faster than tape. $2500 to $4000 for 1Gb. Lots of storage space. The disadvantage is the media can be written only once. This is not a disadvantage if used for image archive only. Media price is $100 to $200.

Read-Write (opto-magnetic or phase change) Laser Disks hold 600 Mb to 1Gb and cost $3000 to $6000. They can be, unlike WORM lasers, rewritten, and their speed is good. Media price is $130 to $250.

Removable cartridge disk (Bernoulli Box) are up to 150Mb and run $1100 to $2500. Media is $90 to $140. Speed is slow, but not as slow as tape.

Floptical Disk holds 21Mb on a 3½ inch diskette. 3 times faster than a standard floppy, 6 times slower than a hard disk. $500 for the drive and $22 for the media.

IBM is now has a 2.4Mb 3½ inch floppy. We have to wait and see if this will become popular in the face of the new floptical drives.

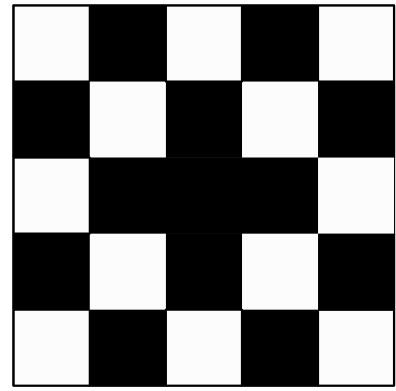
With a storage of 660 Mb, and a price that has dropped to $4000, CD-R (CD-recordable) may become popular storage media.

Display

To display the image we have the system monitor as discussed above. The standard monitor is between 12 and 14 inches. There more and more 15, 16, 17, 19, and 21 inch monitors on the market every day. Super large screen is available also. Another option is to use an LCD (Liquid Crystal Display) screen designed to work with an overhead projector. Also, LCD direct projectors are now on the market. LCD direct projectors, like the LCD systems that work with an overhead projector, are usually limited to the resolution of your monitor and the image quality may not be good if the projected image is of any appreciable size. The images produced are Ok for presentations but are not super quality.

Using a VGA to NTSC converter you can display a computer image on a TV. Besides the bandwidth limitations mentioned above, the converted TV can display only 640 by 440, 640 by 420, or other non-computer standard resolutions.

Film recorders can take the digital information directly from a computer



and expose the film of our choice, normally 35mm slide or print film. 8 x 10 inch color transparencies can also be made using an optional camera back. Color and recording resolution can be outstanding. Currently, film recorders cost $4500 and up. Some older recorders captured the image at the monitor level, limiting the capture resolution to the resolution of the system monitor. Newer recorders are stand alone products, and of much higher quality. 4000 (known as 4K) is good for all but the most demanding applications, say projection on a 30 ft. or larger screen. 4000 is 4096 x 2732 pixels, and is about the limits of quality 35mm film. (Standard film can resolve at 15 to 20 microns). See the information on film scanners in this chapter for more information on image size. See the section The Inside Story for more information about film resolution.

Most film recorders handle vector information. Make sure the one you get can handle bitmap (raster) information in the file format you normally use or can convert to.

Standard film such as Ektacrome 100 may be used on a film recorder with good results, but more expensive professional film is recommended for top quality output. Polaroid can provide a Power Processor that allows on the spot processing of color slides. This is good for proofs, not for the final product. By using a home slide mounting kit you can go from computer to projector in about 10 minutes. Quality is not as good as regular film. Polaroid instant prints may be taken with the proper camera back, and are useful for making a proof photo before using expensive film and processing. Mounting slides on glass mounts will help prevent distortion (slipping out of focus) due to heat during projection. The people who provide scanning service will normally provide prints or slides for $4 to $8 per image, slightly more for color transparencies. Please note that using a good quality projector with a glass, rather than a plastic lens, will help keep image quality at a maximum.

You can also take photographs of your monitor. Distortion from curvature of the monitors screen can be lessened by using a long focal length lens and moving the camera farther away from the monitor. 100mm is a good size. A non-interlaced flat screen monitor will give the best results. Use ISO 100 film and a shutter speed of at least 1/5 of a second. 1/2 to 1 second is better. Geometrical accuracy is a must. Be sure to avoid unwanted room light reflections on the screen. If you need detailed information contact me regarding CRTPIX.TXT, SCRNSH.txt, and VIDMON.THD.

Hard Copy

Next, we will consider hard copy. It is not always practical to carry our computer with us, or to have several hundred people gather around a monitor during a lecture. Therefore, it is often necessary, or desirable, to create a hard copy of our digital image.

Most professional print shops, poster makers, etc., can handle several of the major file formats supported by todays image editing software. Most middle to top-of-the-line software packages can do CMYK (cyan, magenta, yellow, black) color separation needed by professional color printers, saving us the cost of color separation by a service bureau. (See the section The Inside Story for more information). Prices will vary depending on what, and how many, you want. Check with your local print shop about data format and printing prices.

Printing on your own is easy to do, but it is difficult and expensive, to do well. It is also a difficult subject to understand. Again, we will talk about gray-scale images, but most of the information also applies to color images.

Printer prices run $200 to $600 for a B+W dot matrix (up to 360dpi) and $800 to $4000 for B+W laser (300 to 600dpi) printers, with color ink jet (180 to 216dpi) somewhere in between in price. $10,000 for a 300dpi thermal wax full color printer. Do not buy an expensive printer until you have seen the prints of several images you have selected as a test. Continuous tone printers produce the best looking output.

The following allpied to non-continuous tone printers:

Normal thought process would suggest that a 300dpi scan printed on a 300dpi line printer will yield an exact copy of the original, with an image resolution of 300dpi. This is true for B+W, but not for gray-scale or for some color processes. Remember that our scanner, CCD, or other input device may supply data representing 256 shades of gray, or many colors. Our monitor may display 64 or 256 shades of gray. Depending on the type of process used, a monochrome printer, dot matrix, laser, etc., may only supply two shades, ink or no ink. Text or line drawings (true B+W) look good at 300dpi, outstanding at 600dpi. To supply shades of gray between B+W the software, or the printer, must do what is called a halftone process. Professional printers do this by changing the size of the dot printed. Look at a newspaper with a strong magnifying glass for an example. The current crop of printers for the PC, including laser printers, cannot do this. Instead, they use groups of dots and adjust the percentage of black dots verses white paper to approximate gray. (Expect to see vast improvements in printers over the next several years). Figure 7, page , and Figure 8, page , show two different patterns of halftone at 52% black. Since we now need a group of dots to do what one dot used to do, we lose image quality (resolution). A modest 5 x 5 matrix reduces a 300dpi printer to 60dpi gray-scale halftone.

See the section The Inside Story for more information about halftones.

The halftone process can also be applied on the monitor to display more colors (simulated) then the monitor is capable of (such as displaying a 24 bit image on a 256 color system).

A 300dpi laser printer can approximate a 53 line screen. Most newspapers are printed with 85 line screens, and magazines are 120 or 133 lines. For example, PC Magazine currently uses 133 lines.

Relation of DPI to Lines

Halftone Screen Frequency	**Number of Gray Levels**		
	300 dpi	**600 dpi**	**1200 dpi**
30 Lines per inch	101	401	1600
60 Lines per inch	26	101	401
100 Lines per inch	10	37	145
120 Lines per inch	7	26	101

Note 1. The above information is not to be confused with lines as used in photography. That is given in Line Pairs (2 lines) per milameter. Typical film runs 50-100 Lp\mm. See the section The Inside Story for more information about film resolution.

If you see "banding" (streaks) in your print, go to a lower screen frequency. 60 to 50 lines is about the best you can do on a 300 dpi laser printer.

Most image editing software will have several different halftone processes to chose from, depending on the printer and image to be printed. Your printer may have its own halftone options built in. Experimentation is necessary to find the best process for your system.

There are 600dpi laser printers appearing on the market now. Also, some enhancement boards can turn some 300dpi printers into 600dpi printers. Please note that some of these enhancement boards may not improve picture quality, only that of text. Check to see what the real improvement will be seen on the images you work with. True 256 gray-scale 300 dpi laser printers are just hitting the market in the $2500 range. 1000 x 1000 dpi B+W laser printers are running about $7500. Dye Diffusion Thermal Transfer (D^2T^2) printers will give true color photo-realistic prints, but cost from $13,500 to $25,000 for basic units, and can go as high as $40,000.

Modems

Although they are not directly involved in image editing, a short discussion on modems is in order.

Modem is a term derived from modulate\demodulate, which describes the translation of data into form transmittable over a different media. In this case we are taking digital data and converting it to tones, which can be transmitted over the telephone system.

You may wish to use a modem to send or receive information, programs, or images to\from a BBS, a network, or a colleague.

Personal letters are small, programs are medium to large, and images are small, medium, large, and gigantic. Most images are at least large!

If you limit yourself to short, local calls, you can get away with a 2400 baud modem. If you get serious, and you probably will, you should have a 9600 baud or faster modem with "the works". (V.42bis, NMP5, V.42, and V.32bis). See below. The $300-500 you spend on a high speed modem, vs the $100-200 you spend on a low speed modem will be money well spent.

An internal modem will have its own UART (universal asynchronous receiver-transmitter. An external modem will rely on the UART on the serial port in your system. I ether case, make sure the UART is a 16550A chip. The old 8250, 16450, 82450, and the 16550 (no A) UARTs are older, slower, parts which may cause problems at high speeds.

Modem Protocols

Data Compression	Error Correction	Signaling Speed
MNP5. Up to 2 to 1 compression.	MNP1, 2 and 3.	Bell 301. AT&T full-duplex up to 300 bps.
MNP7. Up to 3 to 1 compression.	LAPM. The basis for V.42.	Bell 212. AT&T full-duplex at 1200 bps.
V.42bis Up to 4 to 1 compression.	V.42. Incorporates a fullback to NMP 1,2,3 or LAPM if V.42 is not available. Reduces character size from 10 to 8 bits.	V.21. CCITT full- and Half-duplex up to 300 bps.
		V.22. CCITT to 1200 bps.
		V.22bis CCITT half-

		duplex at 1200 or 2400 bps.
		V.23. CCITT half-duplex up to 800 or up to 1200 bps with a reverse channel of 75 bps.
		V.32. CCITT full-duplex at 4800 and 9600 bps.
		V.32bis. CCITT at 4800, 7200, 9600, 12000, and 14400 bps.

Notes:
1. AT&T is American Telephone and Telegraph.
2. CCITT is the Comite Consultatif International Telagraphique et Telephonique.

Procedures

Read all of the software and hardware documentation and use the Help Function (if available in the software). Some programs, most notably shareware, have valuable information that can help in understanding the many facets of digital image editing.

Proper image capture is the most important first step. Computer, or other, image enhancement may be able to compensate for poor capture technique, but the best results will be achieved if all the steps in the capture\ enhancement are done with the utmost care. Lighting, focus, and exposure are some of the most important.

One of the problems with digital enhancement of bitmapped images is the errors that can be introduced at each step of the process, especially in a color image. For example, an image captured on film, developed as a print, scanned on a scanner, viewed and modified while looking at a monitor, then output to a slide, has many opportunities to be different from the original object. What you see on your monitor can, and probably will, differ from the end (print, slide, magazine, etc.) product. After all, adjusting the contrast or brightness control on your monitor changes only what you see, and not the image data. 100% black printed with a new printer ribbon is different than 100% black printed with a used printer ribbon. Calibration methods, and standards, for computer imaging devices are just starting to be introduced, and should improve rapidly. Use any calibration tools available to adjust your camera, scanner, monitor, printer, or film recorder before you capture or edit an image.

The Pantone, or other, system for color comparison and adjustment is helpful in correcting color, from monitor to printer. Color samples with CMYK values are less than $100 a set.

Set the monitor gamma correction if your software has it. See the software section for more on gamma correction.

If needed, do flat field and dark field compensation before using filters, or doing any image modifications. If you plan on doing this to each image you should set up a batch file to automate the process. For DOS programs you may be able to do this using a DOS batch file and a command line DOS image editing program. For Windows programs you may want to use WINCMD, a free batch program for Windows.

If you can afford the storage space, capture images in all available density and depths. For example, if scanning a color photograph, scan both in color and gray-scale and at several dpi settings. This should include the maximum setting. This will help eliminate errors introduced if size conversion is necessary at a later time.

If storage space is a problem, keep file size at a minimum (except for originals). Images for printing on a linesetter (professional printer) should be captured (if possible) at 1.5 times the final line screen setting of the printer.

The same applies for laser printers. A 300dpi laser printer is about 53 lines. Therefore, images for a 300dpi printer can be captured at 75 or 150dpi.

Images for archive should always be captured at the maximum resolution possible.

Formula for scanning if a change in image size is planed:

final height (or width) divided by the original height (or width)
multiply this by the line screen frequency multiplied by 2
the answer is the minimum scanning resolution.

The following list is a very basic step by step procedure for image capture and editing using a film camera, a flat bed scanner, and a program that includes scanner control. Procedures will vary with software. Much experimentation will be needed to get things just right. For example, which halftone will work best on your printer?

1. Photograph petroglyph using the best lighting, film, camera, etc. Include a scale in for size reference.

2. Processes film and create a print.

3. Calibrate scanner using supplied procedures. Set monitor gamma.
Normally, monitor gamma adjustment will need to be done every 6 months, or if there is a major change in the system or its environment, such as change in lighting. Normally the gamma settings are stored as a file, and then can be loaded, perhaps manually, perhaps as an automatic function. Note: If you readjust the gamma, set the gamma to the normal setting (usually 1.0), and then follow the adjustment procedure.

4. Mount print on scanner, align at 90 degrees to scanner edge.

5. Do a preview scan and set brightness, etc. for best image.
This is important! You do not want to spend your energy making up for problems created by poor procedures. You want to start with the best you can get, and then make it better. Poor input techniques can truncate data (loss of the lightest or darkest information). You may not be able to recover that information no matter how good your software is.

6. Scan the print at maximum resolution and save as an archive image. (You may wish to compress the image to save disk space). Keep notes so you can later enter the filename into a database or catalog system to reference site, date, the original picture the digitized image was created from, and other pertinent information.

7. Use the cropping tool to select portion of image to be edited. (This keeps file size to a minimum). Scan at several different densities. These will be the images you will enhance. Save these images using unique names.

8. Open the image file to be edited. Which of the several different resolution images you use will depend on the final destination of the image.

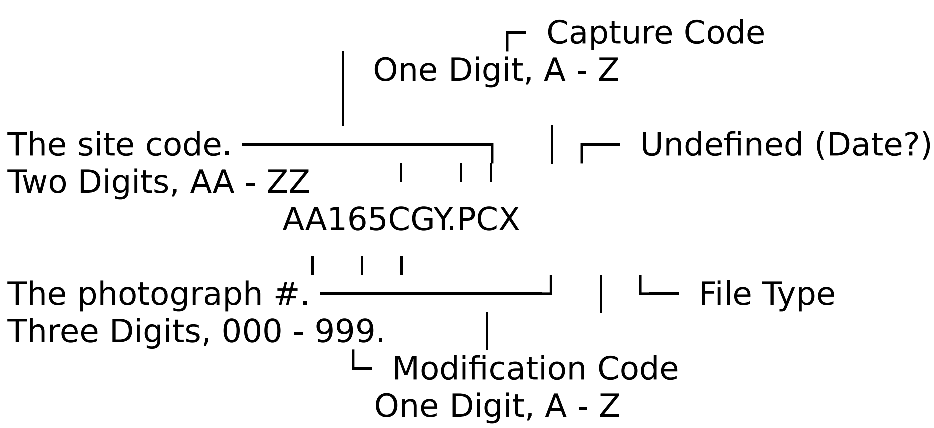
9. Crop the image to size before doing any filtering or adjustments. This will speedup the process, and the filter will work with only the useful data. (This way non-relevant information, such as background and text, will not be used during the enhancement).

10. Use the copy function to copy the image to the clipboard. Clear the existing image. Open a new blank image, much larger than the image you have in the clipboard, and copy multiple images from the clipboard to the new image. Or use the copy or clone tools to do this.

11. Go to work! Leaving the top left image untouched (for reference). Select an image and use the contrast, brightness, sharpen, zoom, invert, and other tools on that image. Don't forget to try CMYK and HSV separation (see the section The Inside Story for details). Move from image to image trying different techniques. Keep notes! A micro cassette recorder will keep you from having to write everything down.

12. Select the image(s) you wish to keep. Using the text function, add text or code to the image to explain what has been done to it. This can done by selecting the image, cutting (or copy) to the clipboard, and pasting to a new picture larger than the original. The text can then be added outside the image area. You may also want to add pointers, circles, or add other marks to explain the image. This is useful for a slide show.

Expanding the image size may be ok for grayscale images, but you may find that expanding the size of a 24 bit image to hold text will make the file unnecessarily large. If you don't need the information for a slide show you may wish to store text in a separate file. The file name should relate to the image name. For example AA165CGY.PCX for the image file and AA165CGY.TXT for the text file. The file name is a code that describes the file and is broken down as follows:



Site code = the site the image was capture at.
Photograph # = the actual photo number.
Capture code = the method and resolution of the image capture.
Modification code = untouched original, modified, pseudo color, etc.

This is a suggestion, not a cast in concrete code. If you have a method that works for you, let me know so I can share it with everyone.

13. Compress (if needed) and save image(s) with unique file name (see above). Add info about the file to a catalog (database). This should be separate from the archive (untouched) catalog.

14. Output the file to printer. Copy the file to a floppy to send to a friend. Send the file to service bureau to be made into to slide or poster. Add the file to a slide show program. Up-load it to your local BBS.

15. Sit back and enjoy your work. Show your work to fiends and family, get other people involved Rock Art!

Things to try

With all the tools at out disposal it is tempting to forget the simple things that make a image useful, and enjoyable. The long time photographic techniques of "dodging" and "burning" are simple but effective tools. Dodging blocks and area from light, making it lighter. Burning exposes an area to light, making it darker. Remember, we are using terms related to photographic paper which gets darker with exposure to light. By darkening the area surrounding the object we wish to study we remove, or at least diminish, distractions. Your program mat not have tools *labeled* as dodge and burn, but this effect can be obtained many ways. Lighten or darken applied wait a tool or brush. Non-linear histogram or gamma adjustment. Masking is another method to use.

When studying an image, try using color (channel) separation as a tool. You may find the results worthwhile. Also try looking a negative of the image.

Adding tint to an image my be a help in bringing out detail. For example, red paint on brown rocks may be hard to see. By adding a light green tint to the image the red paint will darken, allowing us to see the image in more contrast\detail.

Try false color. Make the background blue, and the object yellow.

When using filters, or other time consuming process, try to select a small area that has both the section you wish to enhance, and a section that should not be enhanced, and try this area first. If it looks ok on the small area, undo the filter, then do the full picture. Proper use of "undo" has its use's!

If you wish to add false color to a grayscale image, and yet preserve the original contrast, use the HSV color scheme. You can paint the hue, and the values will care of themselves.

Unless your image editing package has special filters for color images, HSV (HLS, HIS,) color separation is necessary when using filters on color images. If we use a standard filter on an RGB image we end up with a mess. If we first separated the image into its HSV (hue, saturation, and value) components we could filter just the intensity component of the image. HSV is about the same as HIS (hue, intensity, and saturation) or HLS (hue, level, and saturation). When we recombine (merge) the image we have modified the intensity information without distorting the color information. This method works well when sharpening a color image.

This can be used on YCC, YVU, YIQ and other color models. The idea is to adjust the lightness\darkness channel without changing the color. You can even adjust the K (black) in CMYK color space to achieve the same effect.

Color separation can also be used to store large files on small media. Example: If you can't get an image file to fit on a floppy, separate it in to its RGB components, compress each color (losless), and store each on separate floppies.

Note: The conversion from one color space to another, such as RGB to HLS, is never a perfect process. Converting from RGB to HLS and back to RGB will introduce minor changes. Normally, these will be so small they can be ignored. However, the changes are cumulative. (This is true for lossy compression also). Convert back and forth enough times, and there may be noticeable changes in the image. It is a good idea to do all your image enhancement in one sitting, or to start with a fresh copy of the original at each editing session.

Some Image processing programs will allow you to use masks. This may be used to provide a tracing paper like function. Start with a new (blank) image the same size as the image to be traced. Load the image to be traced as a mask, with mask off, but show mask on. In this way you should be able to see the image as a mask, but it should not have any effect. You can now trace around the image and then remove the mask.

Another way to "trace" an image is to have a large (8.5 x 10) transparentcy made, tape this to your monitor, and trace using your mouse. Or you can use a digitizer tablet to trace a photograph.

To enhance an image I like to Copy/Paste the section I wish to use. The section is transferred to the Clipboard using the Copy. I then close the window that contains the old image and open a new, much larger window. The selected portion of the original image is then Pasted back to the new window several times (4, 6, 8, or more). I can then use different enhancement techniques on the copy of the original (leave one image untouched for a base line) and compare results side by side. You can then save the entire "proof sheet" or Copy and Paste the selected images to a new window before saving. This method will save storage space. See Appendix A.

At least one image capture (photographing, CCD camera, etc.) should be done at a 90 degree angle to the object with a scale in view if possible. This allows for measurements.

Scanning density will change the size of the image viewed on the monitor. A 1 inch picture scanned at 75dpi will give about a 1 inch picture on a 640 x 480 monitor. Scanning at 150dpi will enlarge the picture to approximately 2in, without losing quality. Moving up to a 300dpi scan will double the viewed image size again. This is the best way to enlarge or reduce an image. Software reduction or enlargement of a low resolution image can cause unwanted pixelation or image loss. (Remember, the software has to throw away data when reducing an image, and add its own interpolated data when enlarging an image).

For output to a slide, or for professional printing, decide what size the final output should be and then scan the image at the maximum dpi that will allow the image a close fit. Using your editing software, crop off any excess for an exact fit.

If you must re-sample (re-size) an image try to use an even number, such as 2 or 3 times, rather than 1.½, 3⅝, or other partial steps. Re-sizing by partial sizes may cause interference patterns (moiré patterns) to be introduced into the image. Nearest Neighbor scaling is fast but will introduce aliasing. Lanczos3 and Averaging/Linear are much better algorithms.

The resolution your monitor is currently displaying will affect the size of a viewed image. An image will appear 40% smaller when displayed at 1024 x 768 than at 640 x 480. Remember that only the image size you see changes with monitor resolution, not the size of the image file or the resulting output.

An image at approximately 70 dpi on our monitor (640 x 480 pixels on a 14 inch monitor) will be a small image when reduced to 180dpi or 300dpi at the printer. Learn to work with large (on the monitor) images.

For contrast adjustments with the minimum of error due to electronic noise, tape dropout, etc., histogram end-points should be kept to 95% of maximum at both ends of the scale. (This is more important in CCD astronomical work with long time exposures, is not that important in rock art). An Autodensity control is the easy way to set contrast to the maximum with out loosing detail.

Note: a histogram stretch of contrast should be the last enhancement done. I you do this first you run the risk of having other processes modify a pixel to the point of saturation, greater than 255 or less than 0. If you are going to use filters that will do major pixel value modifications to an image, you may wish to compress that contrast using a linear histogram process, filter the image, and then remove the histogram process.

A scale of some sort, marked in inches and centimeters, should be used when photographing an image. This is helpful to understand the actual size of an image, and when pictures have been re-scaled. The sharp, well defined, and edges can be handy if you have to register pictures when doing a multiple image overlay. The scale should be placed so as to appear in a corner of the photograph/image. It should be as close to the subject as conveniently possible when the image is captured. A small hand level can be used to level the scale for reference. If you crop an image in such a manner that you will loose the image of the scale, the image of the scale can be moved to the cropped portion of the image using the cut-and-paste function of your software. Be sure to move the image of the scale to the cropped area before re-seizing an image.

I am trying to find out if there is a standard scale used by scientists in the U.S., and worldwide. If not, or you cannot find one, I have designed one, see Appendix C. One idea for holding the scale during image capture would be to use a microphone stand with several adjustable booms, or goose necks, for placement.

If you are interested in comparing the outline shapes of an image, without the distraction of the inner detail, you can create a "Cameo". Select the portion of the image you wish to use, and use the fill tool to fill the image with a neutral color. You can use the mask to perform the same procedure.

Another way to compare images is to make one a semi-transparent floating image, and then move it over the image you wish to compare against.

Things to Think About

There are many different meanings for the words color, shape, size, gloss, and texture. There are psychophysical, perceived, and cultural definitions.

Psychophysical refers to the measurable qualities of a light or object. A light has qualities including wavelength, intensity, and polarization. A pyramid has measurable lengths and angles. We can measure gloss by the percentage of light reflected, and texture by measuring deviance from a baseline.

Perceived refers to the perceptual phenomenon of our reaction to the light, color, or object. We may call a specific frequency of light red or blue. A texture we see we might describe as smooth. A surface we see as shinny. Or as colors that complement, or clash.

Cultural is the way the color, light, object is viewed (or perhaps recognized) by society. Is pink "a girls color" and blue "a boys color"? Is purple the color of royalty, or is the color of passion? What is too tall, too bright, or too rough here, may be just right some where else.

Some observation on all three of the above:

Visible light-	Color	Wavelength (nm)
	Red	760-647
	Orange	647-585
	Yellow	585-575
	Green	575-491
	Blue	491-424
	Violet	424-380

Peek response of the three types of cones (photoreceptors) in the human eye are at approximately 445, 540, and 570 nm. Therefore we do not have RGB eyes. The 540 and 570 cones overlap, and the 445 is almost independent. A light with a wavelength of 430 nm will stimulate only one cone, 555 nm will stimulate two cones. 495 nm will stimulate all three, but is well away from the peek response points of each.

After light is captured by the cones, more processing takes place, and this processing adds to the way we perceive color. After the cones, it is generally excepted that there are two channels that carry the visual information toward the brain. One is the red-green channel and the other is yellow-blue. Because of this channeling, the attributes of redness and greenness nor the attributes of yellowness and blueness can coexist in a perceived color.

Color sensation is dependent on intensity as well a wavelength. As light gets more intense, both orange and yellow-green approach yellow, while violet and blue-green both become bluer. Yellow, blue, and green seem to be independent of intensity, and are therefore known as "psychological primaries".

Typical optimum viewing distance for a CRT device is that at which the pixels are separated by 1 minute of arc. A typical NTSC TV is 480 x 380 pixels. The optimum viewing distance for this device is equal to seven screen heights. Moving closer will not add detail to the image.

Computer monitors have good brightness and color purity. However they are weak in portraying some colors. These are dark yellow, dark cyan, and dark magenta.

The human eye lacks chromatic correction. This can cause chromatic stereopsis. A red image appears to be in front of a blue image.

The primary colors in a monitor generally have a narrow bandwidth. Much narrower than inks used in printing, or chemicals used in film. Because of the narrow bandwidth, color shift due to changes in lighting conditions will be more pronounced with a monitor than with a print or photograph.

Take and draw to figures <-> and >-<. Label them as #1 and #2. The lines connecting the V shaped ends should be equal length, and should touch the V ends. Now turn the drawing sideways, so the connecting lines are vertical. Ask you fiends in which figure is the connection line longest. Not the total figure length, just the line that connects the V ends. Most people will answer #2 is the longest. Now ask a traditional Zulu, one who still lives in the traditional round Zulu hut. They will say the connecting lines are of the same length. Why the difference in the answers? We who live in modern cites see the objects as corners. Object #1 is corner close to us, with the "walls" going back into the distance. Object #2 is a corner far away, with the "walls" coming toward us. This cultural bias changes our perception of the object. The Zulu, coming from a world of round huts, and natural shapes, is not biased toward perceving these lines as a corner.

Our memory modifies the color we see. Cut two shapes, a leaf and a donkey, from the same shade of gray paper. When viewed against the same background, most people will say the leaf shape is a greener shade of gray then the donkey shape.

Assimilation affects our perception of colors. The lamps we use to light our houses at night are of a yellow tint. We normally do not notice this until we look at an object under the lamp, and then quickly move the object into sun light (bluer).

An RGB computer monitor can not reproduce all the colors a human eye can see. (CIE color space vs RGB color space). In fact, no device that relies on three, or less, sources of color can produce the full range of colors perceivable by the human eye.

Most professional printers, those printers using dots (don't they all?), will print the dots on a diagonal (or the program will). For some unknown reason the eye is least sensitive (that is, we don't see them as individual dots) to dots on a diagonal. Try looking at a newspaper picture from various angles.

Colors appear brighter when contrasted against darker colors. This effect is strongest at the boundary of the two colors. Because of this, a small patch of yellow on a blue background will appear brighter than a larger patch of the same yellow on the same background.

With in limitations, a larger color depth will appear to give the appearance of a higher resolution. That is a 16.7 million color image will appear to be of higher resolution than a 256 color image.

A basic rule of thumb is that the eye can resolve roughly 1/60 of a degree. At a viewing distance of one foot this is about 600 points per inch. Therefore a 8 x 10 inch picture needs to be 4,800 x 6000 to appear completely smooth. (Do not confuse this points per inch with the dpi of a printer. Remember halftone.)

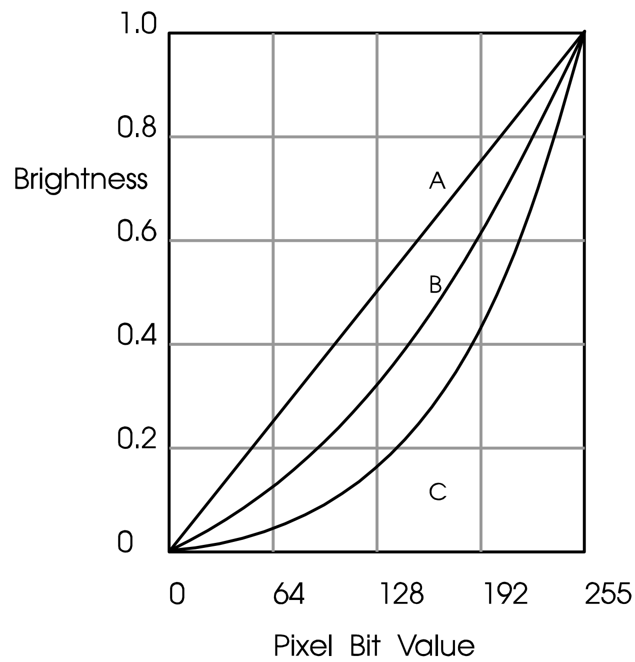
256 gray scale is very close to the max needed to produce the number of gray steps perceivable by the human eye. In technical terms, this is a high enough sampling frequency.

The eye is more sensitive to changes in green than the other primary colors. (Sensitive to hue, saturation, or luminosity? All three? I don't know. Help!)

The eye is least sensitive to blue. Blue is harder to focus on, and so makes a good background color.

Software

Remember that the images we will work with are called bitmapped (raster) images and differ from the vector graphics images that are created



A=Scanner

B=Macintosh

C=IBM and other monitors

by some draw, CAD, or design programs.

Some programs do it all, scanner control (or video capture), filtering, paint, text, printing, file format conversion. etc. Others will do only one or several of the necessary tasks. Most of the programs will do the basic paint and print functions, but usually only the more expensive of the programs contain the filter, histogram, and other functions for serious picture enhancement. The software packaged with the full feature flatbed scanners tend to be upper-middle to top-of-the-line. Software packaged with hand scanners are normally basic to mid-level.

Software vendors tend to add new features, as well as improve old ones with each new version of their software. They often leapfrog over each other for top honors with each new version released. What is best today may be dropped into second place tomorrow with the release of a competitors new version of software. Only data regarding the latest version (revision) of software should be used for comparison of software packages.

Appendix B gives a partial list of commercial software packages to look at. Your local library should have magazines with reviews on some of these programs. Unfortunately, the only way to really see what these programs can do is to test drive them.

Before buying any software, make sure that it will work with your hardware. Do you need Windows or is the program made for DOS? What type of processor do you need to run this software? Will a math co-processor improve speed? How much memory and disk space is needed? Check out the details before you buy!

General Tools

There are many image editing programs on the market, and more being released every day. Some are called image editors. Some are called paint programs, but may still have image editing capabilities. I have not worked with, or test driven, all of them and so can give only general guidelines on what to look for. Programs to compress image data and other utilities will also be addressed in this section.

A good program should have a Gamma Correction setting to compensate for the monitors gamma curve. The brightness change of a pixel on the monitor is not linear with the input. An input value change from 64 to 128 doesn't double the brightness of the pixel. The gamma correction will compensate for this fact. A typical scanner has a gamma of 1.0. A typical monitor will use a gamma correction of 1.6 to 2.2. See Figure 9, page , for a typical gamma curve.

Some software will allow for gamma control of not only the monitor, but will allow for gamma manipulation of the image. This can be used for non-liner adjustment of the image.

Calibration of your input source and output device will save many hours of headaches trying to get the output image to look like the original input. Look for simple to use, effective, built-in or external calibration programs. Having this function as part of your editing package should make life easier. See Appendix B for calibration tools.

The ability to dither the image from color to gray-scale is nice to have. Bayer and Floyd-Steinberg transforms are fast but do not always produce the best output. Burks is slower but better. Stucki is better yet.

Any good program should allow you to input, edit, and output in 24 bit True Color while working on a system with an 8 bit indexed (256 color) or less video display. (This is known as device independent programming. A program that is not dependent on, or limited by, the hardware it runs on). If you have a VGA system you will only see 256 colors or 64 shades of gray while editing, but your final output will be 256 shades of gray or 16.7 million colors when working with a 24 bit image. Some software will do this by displaying halftones, others will translate to 256 color for the display.

An Undo feature will allow you to undo your last change. It is a vary handy tool when testing the effects of different filters. Local Undo will do this for a specific area.

The Magnify Tool (Zoom Tool) is indispensable, and should be included in even the least expensive packages.

A Combine Function (Add Function, Merge Function) is needed to overlay pictures one on top of another so they mix together, rather than cover the image below as happens with a cut and paste operation.

Automatic Registration is a plus if doing image combination (Combine Function, Merge Function) on images that do not align exactly (such as scanned images). It would save time if a section of the image could be selected to set registration instead of analyzing the whole image. See procedures section about using a scale card.

Control over brightness and contrast is a must, with a Histogram and/or Autodensity being a plus. Image Rescaleing is also a must have. A Select Tool (Magic Wand) is a handy tool for selecting images without having to trace them by hand.

See Appendix E for sample histogram\gamma curves.

Format Conversion

The program you select also should be able to handle several bitmapped (raster) file formats. Having an image editing program that can handle several file formats makes image processing much easier. There are many file format conversion programs, but the ability to convert and save in several formats in the image editing program will save you the trouble of jumping between programs to do conversions. See the section on Image Data and Appendix D for file types.

Text

Another of the functions that any image editing software program should do is to allow the addition of text to the image. This allows for identification, explanation, or other data to be included with the image. Besides site and date information, you might want to add some sort of indicator to show which images are untouched and which have been enhanced. For example, a code such as DI Exxx could be placed in the bottom left hand corner of each picture. DI could stand for Digitized Image with the E standing for Enhanced and the xxx being a number referencing the type of enhancement that had been done. A zero would mean that it is the raw (unmodified) image. (If you come up with a good system, please let me know).

Be aware that some text may be added as vector information. This may cause problems at a later time, such as during file format conversion. Be sure to place any text outside the area of your picture that contains valuable data. This way it will not affect filter and editing functions. It night be a wise idea to add a blank area to images just for text, so it can be removed or changed without disturbing the image.

Filters

Look for useful filters and functions such as Sharpen, Remove Noise, Average, Edge Trace, Combine, and Register. Special effects such as Posterize, Skew, Mosaic, Ripple, Bend, Emboss, Blur, and many others only distort the image and are useless for enhancement. The names of the functions may vary from one software package to another. Some of the filters work best on gray-scale so don't count gray-scale out just because you have a color capable machine.

Filters and color separation can be used together on color images. More later.

Color Separation

Color (channel) separation is nice to have for two reasons. First, CMYK (cyan, magenta, yellow, black), RGB (red, green, blue), or HSV (hue, saturation, value) separation can all be helpful in studying an image. By separating the image into its color components, details hidden by other channels may be seen. Secondly, CMYK separation is necessary for most professional printing and we can save money by doing it ourselves (it may take some experimentation to get it right). Also it is a must when using filters (more on this later).

Note: The conversion from one color space to another, such as RGB to HLS, is never a perfect process. Converting from RGB to HLS and back to RGB will introduce minor changes. Normally, these will be so small they can be ignored. However, the changes are cumulative. (This is true for lossy compression also). Convert back and forth enough times, and there may be noticeable changes in the image. It is a good idea to do all your image enhancement in one sitting, or to start with a fresh copy of the original at each editing session.

Compression

As mentioned earlier, there are programs, sometimes with hardware support, to compress files so they will require less space on your disk. Compressed files will transfer quicker via modem or network, and will copy back and forth to your backup media faster. The hardware compression support provides speed, 5 times or better, over the software only compressors. The hardware may be a board plugged into a standard system slot or it may be an option on a video display or video capture card. Price is $600 to $1000. You have to do a lot of images, or do large images at a high compression ratios, to justify the price of the hardware. The software version will normally do the same job and cost around $100.

"Lossy" compression systems will provide compression of 10 to 1 or better with "unperceived" image loss or distortion, and up to 200 to 1 with greater loss of quality. JPEG, MPEG, or JFIF (all from the Joint Photographic Experts Group, or similar group) programs and hardware are now available. Some compression programs run automatically, transparent to the user, others have to be invoked with each use. The problem with these "lossy" programs is that they will most likely use some version of the Discrete Cosine Transform algorithm for compression. These are called lossy compression as they will discard "superfluous" information from the image, based on the assumption the data loss will not be perceived or can be tolerated. The problem here is, what is "superfluous", and what is valid information we need when doing image enhancement? A 10 to 1 or 15 to 1 image compression may be fine for working files, but archive files (originals) should not be compressed with a lossy compression system. Lost data is lost forever!

The up side to JPEG is that it is an intelligent process based on the way the eye gathers information. It reduces colors, and retains luminosity (light\dark) relationships. If used correctly it can be an effective tool for image compression.

There is a losless version of JPEG.

Beware! At this time JPEG (pronounced 'jay-peg') is a well defined compression standard, but it is not a file format standard. Some companies are producing software and hardware on preliminary data and proprietary file formats. This can cause compatibility problems when transferring images between systems with different hardware, or software, JPEG systems. The JFIF format is designed as an interchange format and hopefully will become a stable platform to work from. (But, because it is lossy, it will still be of little use to us).

There is a new compression system based on Fractal (Mandelbrot) mathematics that may be of interest. It is called Fractal Image Format and Fractal Transform Resolution Enhancement. Fractal is not a losless system, but it "looks better" than other lossy systems, at least that is what is claimed. If you must have large compression ratios, this is something to look into.

Lossy compression is good for thumbnail copies, icons, and other quick and dirty uses.

Note: Lossy compression is additive with each use. Each time you save a file using a lossy compression program, more information is lost.

Other types of software, not specifically targeted at image files, can also be used for file compression. Stacker, Double Disk, and others give a 50% or better file size reduction across your entire disk, not just image files. You may lose the ability to use some of your utility programs, such as the Norton Utilities, so a trade off may be necessary to use these types of programs. If you wish to retain the use of utilities such as disk defragmenters and data recovery programs, it is best to use a compression program targeted toward individual files, and not one that works on the entire disk.

Other programs, such as PKzip, Scrnch, and Arc (all shareware), and LHA (free), can compress some files by approximately 20% to 50%. Currently, they must be invoked each time you want to use a compressed file. For example, you have to un-compress a needed file, use the file, and re-compress the file when done. Most of these programs can compress a group of files into one archive file. Using this method you could compress all of your image files into one, or more, compressed files. For example, all image files from site "A" could be compressed into one file at a possible savings of up to 50% in disk space over the individual files. The same for site "B", and so on. This allows you to keep files separate by site, image type, or other distinction, and only compress and de-compress once per group. The price is good at free to $50, and you can use them on more than just image files. You can add in text files related to the image files (site reports, notes, etc.) for even more space savings.

Diet version 1.20 (or better) is a free program that can give 20% to 50% file compression on some files, including some types of image files. It also can be run memory resident so it will de-compress and re-compress files automatically during use. Unfortunately, it is from Japan and therefore it is a bit hard to understand the manual (written in English) and to obtain support. It also will cause some programs to error when trying to read compressed files. Slim (shareware, $50) is like Diet but works on a wider range of files with fewer problems.

DOS 6 has built in disk compression. Though it will not do much for already compressed images, it will compress other files, leaving you more room for images. Easy to install, works fast.

For those who tend to rush into things, read the software manual and heed the warning not to compress certain DOS, Windows, and TSR (terminate and stay resident) files.

Inexpensive Programs

There are many other programs that can be handy. The following is a short list of the many programs that you may find useful. They are shareware and therefore have a vary reasonable price. Most will work with GIF file format, some work with several formats.

Gifdesk can display page after page of miniature GIF images allowing you to look for a specific picture. GDS (Graphics Display System) makes it easy to display a slide show, picture after picture, at selected intervals. GDS for Windows is an excellent slide show and catalog program that will handle many file formats. PixFolio is even better, showing thumb nail sketches, and able to handle many file formats. Combined with text these programs can be an easy way to present a show or tutorial. A GIF viewer will allow your fiends with computers to view, but not edit, images you have sent them via modem or on floppy. Cshow, Vuimg, and Picview are some of the better ones that I have seen. There are also programs, such as Gifexe, to make GIF files display themselves by simply typing the file name. Price is less than $50 a program.

Picem is a good DOS viewer and slide show program. Works with PIC, PCX, and GIF formats. Public domain.

Shareware GIF viewers/file format converters for Windows include Paintshop (Paint Shop Professional is very good), Show GIF, and Wingif.

The GIF format is well documented and supported by many programs. It is useful for its compatibility with many current system capabilities. There are GIF programs for the Macintosh, Amiga, and other computer systems. The drawback of the GIF format is that it supports a maximun of 256 simultaneous colors, and most of your high quality work may be in 24 bit PCX or TIF format. Please note that this is 256 colors, selected from a palette of 16.7 million. Therefore, GIF has the potential to show as many different colors as a 24 bit image, but it can only show 256 of them at a given time. File conversion is easy, but you could end up with duplication of files (taking up disk space) to support two, or more, different formats. Even though many programs have GIF as part of their name, some will work on several different file types.

Infoplus (Info+) is a free program that will "look" inside your system and tell you what hardware you have. Some commercial programs, such as The Norton Utilities, etc. come with similar programs. Use them to find out what you have without having to open your system and trying to decipher the hardware.

Image Alchemy ('286 or better required) is a DOS file format conversion program that covers a wide range of file types including conversion from some non-IBM formats such as Macintosh and Sun. It contains good information regarding file formats and supports conversion between many file types. It is not user friendly. About $80.

A disk cataloging or archive program can be of use. A catalog program will help keep track of images stored off-line. Stowaway (shareware) will allow you to archive data in groups and easily restore this data. You may wish to save and recall groups of images sorted by site, age, or by other criteria.

Getting Started

To get started at a reasonable price, here are some software and hardware suggestions. Most programs mentioned here are shareware or demos. These are some of the least expensive ways to start.

I MUST START BY GIVING APOLOGIES WHERE THEY ARE DUE. I HAVE NOT, CAN NOT, TEST AND EVALUATE ALL THE LATEST SOFTWARE, HARDWARE, FREEWARE, ETC! SOME PEOPLE GET PAID TO DO SO! SOME DO A GOOD JOB, OTHERS DO OK ...
I AM ONLY TRYING TO GIVE GENERAL INFORMATION, AS I SEE IT, AMID ALL THE OTHER *MASSIVE* INFORMATION I AM TRYING TO SORT OUT AND FILTER INTO A FORMAT I HOPE MY READERS CAN BENEFIT FROM.

If you name wasn't included, if I talked bad about an old revision of your program, if I missed something good, I do not believe it was intentional. Yes, it is "intentional", in the fact that I wrote the words, but I hope that I hold no "brand loyalty", that I hold no bias. I "calls them as I see them, and I don't claim to look too close". I you have a compliant, you can:

1> Tell me about it. I'm not that hard to find. I case you haven't noticed, I solicit input!

2> Send me a evaluation copy, and point out the good things I missed.

3> Do nothing.

Image Editing Programs

The first three image editing programs should be in everybody's tool box. Good programs at a great price.

Improcess (version 4.0 or better) is one to get. It runs on a PC XT (8088) or better! It is a DOS shareware program that will allow you to view, edit, and filter gray-scale or color images for only $25.

For '286 or better system, GSIP is another great DOS shareware bargain at $12. Grey-scale editing of 8 bit PCX files. Includes filters and false color.

OVIP1 version 1.0 looks hot, but needs to have filters added to be really useful (DOS, shareware $20). Gray scale-only. Nice Macintosh like multi-window interface.

For DOS users, look for a public domain program called Piclab. Not easy to use, but will do some editing, and the price is right. Desktop Paint 256 (DOS, shareware) is a good paint program at $35. No filters.

Windows users, look at the above, and also get the demo versions of PhotoFinish by ZSoft, Image-In-Color Professional by Image-In, and PhotoStyler by U-Lead (now owned by Aldus).

Windows Paintbrush (comes with Windows) has minimal capabilities to view, edit, and print images in PCX and BMP format.

Graphics Workshop for DOS and GWS for Windows are worth a try.

Piclab is a good Windows program, give it a try.

Compression

LHA and Diet are public domain (free) programs if you need to compress files to save room on disks or to cut down on modem time. These will work on most image and data files. Will not help on image files that are already compressed, such as GIF files. PKzip is shareware, and the most popular compression software. All are DOS programs.

Miscellaneous

Look at Image Alchemy (DOS, shareware) for file format co8nversion. It is one of the best around. Not easy to use, but will handle file formats from several computer manufactures (shareware, $80, a '286 or better system is required).

Graphic Workshop (DOS, shareware) is a good, and easy to use, viewer and file format converter.

Graphics Workshop for Windows (shareware, $50) is a good file viewer and format conversion program.

Picem is a public domain DOS viewer and slide show program for PIC, PCX, and GIF formats (8 bit indexed only, no 24 bit).

Pman is a Windows image editor. First version is lacking custom filters. Wait for version 2.

GSIP is a good grayscale editor.

There are several public domain JPEG compression programs available on local BBS's.

GIF viewers abound. Take your pick. Cshow is one of the better ones for DOS.

The screen capture in Paint Shop Pro (Windows, shareware, $50, a good viewer with some editing and enhancement) will allow you to capture images from your monitor and transfer them to other Windows programs, or save them as files. Works great with programs that do not have print capabilities.

If you need to transfer data from a Macintosh to a PC, copy the file in TIFF or PICT (uncompresed) to a floppy disk formatted for DOS using the Macintosh Super Disk and Macintosh DOS Mounter.

From the PC end you can store the image on a Macintosh formatted disk and read it on the PC using Macsee (DOS, shareware $35). Use Image Alchemy (shareware, '286 or better required) to convert to the needed file format.

If file size is a problem, and the systems are close togther, you can link the systems port to port, using a "null modem cable", and then use a communacation program to transfer the files.

WINCMD is a free batch language that will allow you to automate repetitive procedures when using Windows programs.

Systems

Below is a chart showing several systems and their approximate used/new costs. These systems are balanced for cost/performance, but in most cases can be updated piece by piece. For example, a 24 bit high resolution video system can be added to a '286 system. A '286 motherboard (system board) can be replaced with a '486 motherboard while keeping the existing video, box, power supply, disk, etc.

If you currently own a XT (8088), use it. If you are buying a system, don't settle for less than a '386. A DX or DX2 processor is preferable over an SX. Spend your money on high quality video and scanner (or other input device) first, and think about processor power later. Trade processing time for quality! GIGO stands for garbage in, garbage out. A slow CPU will not distort an image. A cheep input device will.

Systems running Windows need at least 4MB of memory to run well.

Image Editing Systems

	Low End	**Mid Range**	**Mid Range**	**High End**
CPU	8088 (XT)	80286 (AT)	80386DX or SX	80486DX, SX or better
Speed	4.7 or 10Mhz	10 to 20Mhz	25 to 33Mhz	33Mhz or better
Memory	1Mb (max)	2 to 8Mb	8 to 16Mb	16Mb or more
Video	SVGA see note 1	SVGA	SVGA, 15, 16, or 24 bit, high resolution	24 bit, high resolution
Co-processor	Add later	Add later	Yes, add to DX or SX CPU's	Built-in on DX CPU's, add to SX CPU's
Hard Disk	20Mb or better	60Mb or better	100Mb or better	200Mb or better
Floppy Disk	360Kb or better see note 1	1.2 or 1.44Mb	1.2 and 1.44Mb	1.2 and 1.4Mb
Operating System	DOS 3.1 or better	DOS 3.1 or better, Windows 3.1	DOS 5.0 and Windows 3.1 or OS/2 2.0	DOS 5.0 and Windows 3.1 or OS/2 2.0
Storage/ Backup	Floppy	Floppy	Floppy, tape and/or floptical disk	WORM or R/W laser disk and/or Tape
Software	Improcess, Piclab, and\or low priced commercial software	Improcess, OVIP1, GSIP and\ or low to mid priced commercial software	Improcess, OVIP, GSIP and\or mid to high priced commercial software	Improcess, OVIP, GSIP, and the top of the line commercial software
System Cost	$400/$1000	$800/$2000	$2000/$4000	$8000 and up

used/new				
Add:				
Mouse	Yes	Yes	Yes, and a trackball.	Yes, and a trackball and a pressure sensitive tablet.
Output Device	9 or 24 pin dot matrix printer	24 pin dot matrix printer	24 pin dot matrix or laser printer	Laser printer, film recorder
Input Device	Hand scanner	Hand or flatbed scanner	Flatbed scanner or video capture board.	Flatbed scanner, video capture board, or film scanner

Note 1. System BIOS (firmware chips on the motherboard) may need upgrade. See your dealer.

The Inside Story

Image Filters

This section will discuss the actual operations taking place when a filter is applied to an image, the use of custom filters, several methods of halftone, etc, etc, etc. This is the "inside story".

The information in quotes (in this section) is taken from FILTERS.DOC as found in IMG_FLTR.ZIP (by John Wagner of Improces fame?). He gives credit for most of the quoted information to the program CONVOLVE. I quoted this material because I could not have said it any better. Some of the filter names come from the same document.

The examples I show here were generated using a Louts 1-2-3 template I developed to test various filters. It's yours, free for the asking.

Another way to study the effects of a filter on an image is to use a paint program to paint an image with known values. Apply the filter, magnify the image, and use the Eye Dropper tool to read the new pixel values.

Now, down to business. When we change the brightness of an image we apply what is called a *point process*. By adding a value to each pixel, we brighten the whole image by that value. Each pixel is treated as a distinct individual, not connected to its neighbors. When applying a filter, we will be using an *area process*. We will analyze an area when determining what changes to make to our image. This type of filter process is also called *convolution*. (If the algorithm is used to change the arrangement or position of the pixels it is called a *geometric process*. If our program changes the pixel value based on more than one image it is called a *frame process*).

Please note that this first discussion is limited to the types of filters you normally find in the image editing programs mentioned in this text. There are others, including fast Fourier transform, Maximum Entropy Deconvulation, and the new Fractal Transforms. I will try to cover these at a later date (As soon as I can learn about them, that is. Any help, or information, will be appreciated).

To filter an image using convolution, we first target a single pixel of the original image. We then evaluate the pixels around the target pixel, and then target pixel itself. The evaluation process will generate a new value for the target pixel, which is then written to a new image. We then select the next pixel in the original, unmodified, image and evaluate the area around it. Using this data we again generate a new value and write this data to the new image. In this way, we paint a new image, pixel by pixel, using new data generated from an area in the old image.

The filter is defined by a kernel, usually a 3 x 3 matrix, but may be 5 x 5, 7 x 7, etc. An odd number is used so there is a center, or target, value.

Example:

```
    P P P   Where:
    P T P       P = pixel to process.
```

P P P   T = target pixel, also processed.

To sharpen an image, the kernel may look like this:

-1 -1 -1
-1 9 -1
-1 -1 -1

The kernel is "placed" with the center (in this case the number 9) "over" the target pixel in the original image. Each pixel, including the target is then multiplied, point by point, by the value in the kernel "above" it.

Original Image Data	x	Kernel Values	=	Resulting values used to generate a new pixel value
37 44 40		-1 -1 -1		-37 -44 -40
37 53 31		-1 9 -1		-37 477 -31
37 58 58		-1 -1 -1		-37 -58 -58

We then add all the new values together. The result is 135. This becomes the new value for the target pixel in the new, not the original, image.

This new value can be used as is. We may add a number to the new pixel value (Bias). Divide it by a number (Factor). Multiply by a number (Boost or Product). We may modify this value using other criteria, or use a combination of above, before writing data this to the new image. In this way, we can modify the effect the filter has.

So, the new pixel value equals the sum of the products of the kernel and the original values, divided by the Factor, multiplied by the Product or Boost, then add the Bias value. Some programs may have only one, or maybe none, of the modification controls (factor, boost, or bias).

If we have a color image, the same process is done for each of the three RGB channels per pixel.

We then move the kernel to the next target pixel and repeat the process. As we need to have the target pixel surrounded, the pixels on the outer most edge of the image will not be processed. The outside edge may be left untouched, or may be filled with the nearest neighbors value, depending on the program.

If we used a Laplacian (high pass) filter, and added the original value back in, we would get the same 135 for our new pixel value.

A Laplacian kernel:

-1 -1 -1
-1 8 -1
-1 -1 -1

"High-pass filters accentuate the high-frequency details of an image while leaving the low-frequency content intact. High pass filtering is used whenever objects with high spacial-frequency content need to be examined. The higher-frequency portions of an image will be highlighted while the low frequency portions become black. The use of High pass filters may highlight images at the expense of adding noise to the image. High frequency in images can be found by looking at edges of objects. Edge enhancement of an image is possible with the use of High-pass filtering."

High pass spatial filters:

HIGHPASS191:

```
-1 -1 -1
-1  9 -1
-1 -1 -1
```

HIGHPASS150:

```
 0 -1  0
-1  5 -1
 0 -1  0
```

HIGHPASS125:

```
 1 -2  1
-2  5 -2
 1 -2  1
```

Without adding the original pixel value back in, the Laplacian kernel functions as a non-directional edge detector for positive or negative brightness slopes. If you add the values of the kernel together you get a value of 0. This will cause areas of the same color to become black and the areas of contrast to be shown as the difference.

Original image #1:

```
100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100
150 150 150 150 150 150 150 150 150 150
200 200 200 200 200 200 200 200 200 200
200 200 200 200 200 200 200 200 200 200
200 200 200 200 200 200 200 200 200 200
200 200 200 200 200 200 200 200 200 200
```

After Laplacian Filter:

```
000 000 000 000 000 000 000 000
000 000 000 000 000 000 000 000
000 000 000 000 000 000 000 000
000 000 000 000 000 000 000 000
000 000 000 000 000 000 000 000
150 150 150 150 150 150 150 150
000 000 000 000 000 000 000 000
000 000 000 000 000 000 000 000
000 000 000 000 000 000 000 000
```

Since our monitor cannot show negative numbers, any negative numbers generated during the process is shown as 0 (black). Anything larger than 255 is truncated to 255 (white).

Notice how this filter causes any area where there is little or no change, known as low frequency, to be removed (black). The areas where there is a large change (known as high frequency) is brightened. We now see only the area of transition, and have removed the areas of continuous tone. In this example, the image started as black and light gray separated by a one pixel line of medium gray. After filtering it becomes black with one line of gray near where the black\gray transition occurred. Remember, the outside edge cannot be properly processed, and so is not shown.

Note the position shift at the transition of color. A purest will take measurements on the originals, unmodified, image only.

If we pass our sharpening (HIGHPASS191) kernel over the same original we end up with:

```
100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100
000 000 000 000 000 000 000 000
150 150 150 150 150 150 150 150
255 255 255 255 255 255 255 255
200 200 200 200 200 200 200 200
200 200 200 200 200 200 200 200
200 200 200 200 200 200 200 200
```

Here our light gray (100) and the dark gray (200) remain, and the transition (150) becomes tri-color. The sharpen filter left the original data in the areas of little or no change, but gives us a black\gray\white stripe at the transition.

If the transition in the original image was vertical, instead of horizontal, it would still enhanced. So far the filters discussed are non-directional.

"Laplacian edge enhancement differs from the other enhancement methods since it is omni-directional. It highlights edges regardless of direction. Laplacian edge enhancements generate sharper edge definition than do most other enhancement operation. Additionally, it highlights edges having both positive and negative brightness slopes. All regions in the image which illustrate a rapid change will be accentuated and non-varying regions attenuated."

Four Laplacian Filters:

```
  LAP1:     LAP2:     LAP3:     LAP4:
  0  1  0   -1 -1 -1  -1 -1 -1   1 -2  1
  1 -4  1   -1  8 -1  -1  9 -1  -2  4 -2
  0  1  0   -1 -1 -1  -1 -1 -1   1 -2  1
```

We also can make filters directional. For example here are two versions of horizontal filters. For the next set of examples, we will use a new original image.

"Using Gradient edge detectors allows you to specify a direction for the edges other than just horizontal or vertical. Diagonal edges can be specified as well. This is accomplished by using directions to indicate the exact direction of the edges desired. If a positive slope in the direction of the filter exists, a light-colored pixel will be placed in the resultant image. For example if the East Kernel is used, a light-colored pixel will be placed in the output image if there is a transition from black to white in the east (left to right) direction of image. Constant regions will be attenuated while regions of high frequency change will be accentuated."

Original Image #2:

```
100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100
100 100 100  37  44  40 100 100 100
100 100 100  37  53  32 100 100 100
100 100 100  37  58  58 100 100 100
100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100
```

A South Kernel -1 -1 -1 yields:

```
 1 -2  1
 1  1  1
```

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	93	0	121	0	0	0
0	0	0	194	157	157	26	0	0
0	0	63	105	147	84	42	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

A different South Kernel -1 -1 -1 yields:
0 0 0
1 1 1

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	14	32	32	18	0	0
0	0	63	110	178	115	68	0	0
0	0	63	105	147	84	42	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

A North Kernel 1 1 1 yields:
1 -2 1
-1 -1 -1

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	63	119	179	116	60	0	0
0	0	0	180	167	179	8	0	0
0	0	0	65	0	57	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

A different North Kernel 1 1 1 yields:
0 0 0
-1 -1 -1

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

0	0	63	119	179	116	60	0	0
0	0	63	110	178	115	68	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

The vertical kernels:

East GRADE:	West GRADW:		East	West
-1 1 1	1 1 -1	Or	-1 0 1	1 0 -1
-1 -2 1	1 -2 -1		-1 0 1	1 0 -1
-1 1 1	1 1 -1		-1 0 1	1 0 -1

The diagonals kernels:

North East South East North West South West

GRADNE:	GRADSE:	GRADNW:	GRADSW:
1 1 1	-1 -1 1	1 1 1	1 -1 -1
-1 -2 1	-1 -2 1	1 -2 -1	1 -2 -1
-1 -1 1	1 1 1	1 -1 -1	1 1 1

Those readers that haven't fallen asleep yet will note that a 3 x 3 matrix is biased toward 45 degree angles. Using a 3 x 3 filter limits us to a 45 degree maximum effective angle. I use the term "maximum effective angle" to describe the angle at which the filter is tuned for. Using a 5 x 5 or greater matrix will allow us to design filters tuned for angles smaller than 45 degrees. If you had to read this twice, put bookmark back 3 pages, take a break, and come back later. This, like much else in this paper, can get realy hard to absorb, understand, care about, like ...

Your back! So soon? Ok, here we go ...

A blurring filter (when used with a factor value of 9):

1	1	1
1	1	1
1	1	1

"Low-pass filters leave the low-frequency content of an image intact while attenuating the high frequency content. Low-pass filters are good at reducing the visual noise contained in an image. Noise is garbage found in image that does not pertain to image. Low frequency areas in an image is where the color of the pixels vary slowly or remains constant."

LOWPAS10:
0.1 0.1 0.1
0.1 0.2 0.1
0.1 0.1 0.1

LOWPAS19:
0.1111 0.1111 0.1111
0.1111 0.1111 0.1111
0.1111 0.1111 0.1111

3LOWPAS486:
0.0625 0.125 0.0625
0.125 0.25 0.1258
0.0625 0.125 0.0625

"Shift and Difference Edge Enhancement. As the name implies, these filters enhance image edges by shifting an image pixel and then subtracting the shifted image from the original. The result of the subtraction is a measure of the slope of the brightness trend. In an area of constant pixel intensity, the subtraction will yield a slope of zero. Zero results in black pixel values. In an area with large changes in intensity, an edge, for example, the subtraction will yield a large value for the slope, which will become a light colored pixel. The larger the difference in intensities, the lighter the pixel."

SDHVEDGE: Shift and difference horizontal and vertical and edges.
-1 0 0
0 1 0
0 0 0

SDHEDGE: Shift and difference horizontal edge.
0 -1 0
0 1 0
0 0 0

SDVEDGE: Shift and difference vertical.
0 0 0
-1 1 0
0 0 0

If we were to use the SDHVEDGE filter on the original image #1 the result would be:

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
50	50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50	50
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

If we were to use the SDHVEDGE filter on the original image #2 the result would be:

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	16	0	60	0	0
0	0	0	0	21	5	68	0	0
0	0	0	0	63	42	42	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

Many filters are built-in and non-adjustable. Not all are useful in the type of work we are doing. They may include, and may function like this:

Average. The target pixel value is replaced with the average value of the neighboring pixels. (Does this include the target pixel value also? I don't know). So, the new target pixel value is the sum of the pixel values divided by the kernel size (a 3x3 kernel would divide the total by 8 or 9 depending on whither the target pixel value is included).

Median. The target pixel value is replaced with the median value of the surrounding pixels. So, the new target pixel value is the value of the middle, after being sorted by value, of the test pixels. If we had a 3x3 matrix in our original image with the values 37, 44, 40, 37, 53, 31, 37, 58, and 58, they would be sorted by value as 31, 37, 37, 37, 40, 44, 53, 58, and 58. The value 40 is in the physical center of the 9 values, and so it would be used as the new target pixel value. It blurs an image.

Maximum. Analyzes the pixels surrounding the target pixel and uses the highest value found as the new target value. Expands the highlights and reduces the shadows.

Minimum. Same as the maximum but uses the lowest value. Expands the shadows and reduces the highlights.

Equalize. This is the same as a histogram stretch. The darkest color is set to black, the lightest color to white, and spreads the others in between.

Neutral. For color images only. Removes color shift. A target pixel whose color is close to neutral (gray) is changed to neutral. Example: a pixel that is almost gray may contain a slight amount of red. This red would be removed converting the pixel to gray. This filter should include a threshold setting.

When using filters on color images it is often useful to work in something other than the RGB color space. For most people, intensity defines the edge of an object, and color fills the area defined by the edge. As an example, lets say we have a red square on a blue field. We want to filter the image so there is the minimum shock in the transition from one color to the next. We want two distinct colors but we want them to blend together as much a possible. If we were to use a filter on the RGB image we would end up with a purple fringe around the square. If we first separated the image into its HSV (hue, saturation, and value) components we could filter just the intensity component of the image. HSV is the same as HIS (hue, intensity, and saturation) or HLS (hue, level, and saturation). When we recombine (merge) the image we have a smooth intensity transition without the purple edge. This method works very well when sharpening a color image.

Now that you know how convolution processing works, be sure to find out what each filter you use actual does. If the documentation that came with a program doesn't explain in detail a filter, demand that the manufacture supply such information. I wouldn't have had to write this section, at least not all of it, if more manufactures provided decent documentation.

Various Filters:

"Edge Detect Light"	"Edge Detect Medium"	"Edge Detect Heavy"
0 1 0	-1 -1 -1	1 -2 1
1 -4 1	-1 8 -1	-2 4 -2
0 1 0	-1 -1 -1	1 -2 1
Factor 1	Factor 1	Factor 1

"Sharpen Light"	"Sharpen Medium"	"Sharpen Heavy"
1 -2 1	0 -1 0	-1 -1 -1
-2 5 -2	-1 5 -1	-1 9 -1
1 -2 1	0 -1 0	-1 -1 -1
Factor 1	Factor 1	Factor 1

Note: The "Sharpen Light" appears to be mislabeled. I believe it to be an edge detect!

"Enhance Detail"	"Focus"
0 -1 0	-1 0 -1
-1 10 -1	0 7 0
0 -1 0	-1 0 -1
Factor 6	Factor 3

So far this discussion a focused on 3 x 3 filters, but most software will allow for 5 x 5, or even 7 x 7. How do you chouse the size to use? Using the magnify tool and examine the edges you wish to sharpen. Try and determine the width of the fuzziness. The width of the filter, minus one, divided by two should be close to the width of the fuzzy area. If you wish to sharpen a one pixel wide area, use a 3 x 3 filter. A two pixel sharpen needs a 3 x 3 or a 5 x 5 filter.

Vertical Edge Detect	Horizontal Edge Detect
0 -1 0 1 0	0 0 0 0 0
0 -1 0 1 0	-1 -1 -1 -1 -1
0 -1 0 1 0	0 0 0 0 0
0 -1 0 1 0	1 1 1 1 1
0 -1 0 1 0	0 0 0 0 0

All filter contributions gratefully accepted! Oil, air, drain, and noise filters have been done already ... be useful, be inventive, be creative. Be alert, the world needs more lerts...

Halftone or Dithering

Normal thought process would suggest that a 300dpi scan printed on a 300dpi line printer will yield an exact copy of the original, with an image resolution of 300dpi. This is true for B+W, but not for gray-scale or for some color processes. Remember that our scanner, CCD, or other input device may supply data representing 256 shades of gray, or many colors. Our monitor may display 64 or 256 shades of gray. A monochrome printer, dot matrix, laser, etc., can only supply two shades, ink or no ink. Text or line drawings (true B+W) look good at 300dpi, outstanding at 600dpi. To supply shades of gray between B+W the software, or the printer, must do what is called a halftone process. Professional printers do this by changing the size of the dot printed. Look at a newspaper with a strong magnifying glass for an example. The current crop of printers for the PC, including laser printers, cannot do this. Instead, they use groups of dots and adjust the percentage of black dots verses white paper to approximate gray. (Expect to see vast improvements in printers over the next several years). Figures 7 and 8 show two different patterns of halftone at 52% black. Since we now need a group of dots to do what one dot used to do, we lose image quality (resolution). A modest 5 x 5 matrix reduces a 300dpi printer to 60dpi gray-scale halftone.

Also note that our 5 x 5 matrix will allow only 26 shades of gray. (actually 24 shades plus white and black). An 8 x 8 matrix is needed for 65 shades and a 16 x 16 matrix for 257 shades.

The pattern used for the halftone process can create banding or distortions in the printed image depending on the pattern of the matrix itself. Bayer, Hex Bayer, and Fatting halftone transformations are fast but may not produce a decent print. Diffused and Enhanced (like Diffused but with better contrast) may be the better choice. But you need to try all available options to see what works best for you.

The halftone process can also be applied on the monitor to display more colors (simulated) then the monitor is capable of (such as displaying a 24 bit image on a 256 color system).

A 300dpi laser printer can approximate a 53 line screen. Most newspapers are printed with 85 line screens, and magazines are 120 or 133 lines. For example, PC Magazine currently uses 133 lines. Most image editing software will have several different halftone processes to chose from, depending on the printer and image to be printed. Your printer may have its own halftone options built in. Experimentation is necessary to find the best process for your system.

Now, we will discuss some of the various methods used, and their respective benefits and drawbacks.

As usual I will use 256 gray-scale as an example, but this information applies to color. Also note that halftone applies to monitors and scanners, as well as printers.

All discussion is directly applicable to devices that use square pixels (dots) arranged in rows and columns. This includes 300 x 300 laser printers and VGA monitors. Most dot matrix printers, and EGA or CGA monitors, are not square. These devices, and those that use diagonal patterns, may show some distortion if square device algorithms are used. (Note: a typical 24 pin dot matrix printer head is 3 wide by 8 high. A properly designed printer with a proper driver should be able to print a square area).

Dithering is a one way process. Images scanned with dithering are less than the original, and usually cannot be edited. Don't bother to save dithered files.

Three things that determine the method of halftone most useful to us are artifacts, errors, and time.

We can dispense with time right now. The better the dithering process the more time it takes to process an image. Patience is a virtue.

We have mentioned previously that the pattern contained in the original image and the pattern used by the halftone process can "create" an unwanted pattern in our output image. This is one problem we need to avoid.

Also we need to select a process that will give us the least error. We are trading spatial resolution for color resolution. A 5 x 5 matrix will produce 26 possible shades. If we start with 256 shades, we have a color reduction of 9.8 to 1. (And a spatial reduction of 5 to 1).

If we were to output our 256 shade image in strict black and white, any value from 0 to 127 would be black, and values of 128 and up would be white. Values of 2 or 7 changed to a 0 have small error. With a value of 100 or 117, we have a massive error between the original and the output image.

One of the simplest dithering methods is known as Random dithering. This method first generates a random number (white noise) between 0 and 255. This number is compared to the target image value. If the random number is smaller than the target value, then a 0 (black) value is used. If the random number is greater than the target value, a 255 (white) is used. Fast, simple, prone to large errors. If we process an image consisting of a solid gray, the resulting image would be "salt and peppered" with white and black dots. A typical output usual appears to be "salted", much like "snow" in a TV image.

The resulting image, though full of noise, will be free of artifacts.

The next step up is patterning (cluster), which is the method described earlier in the paper. This involves using a matrix of dots. I will not go over this information again. We do need to hope that the programmer was smart enough to not use patterns that readily create artifacts, such as:

```
0 0 0   0 0 0   0 0 0   0 0 0   0 0 1   0 1 1   1 1 1
0 0 1   0 1 1   1 1 1   1 1 1   1 1 1   1 1 1   1 1 1
0 0 0   0 0 0   0 0 0   0 0 0   0 0 0   0 0 0   0 0 0
```

These patterns are bound to generate artifacts. Any pattern that repeats on a horizontal, vertical, or diagonal is to be avoided.

Interesting enough, in times past, once a target was used, it should be included in the next successive value. In diagram form:

```
0 0 0   0 0 0   0 0 0   0 0 0   0 0 1   0 1 1   0 1 1
0 0 1   0 1 1   0 1 1   0 1 1   0 1 1   0 1 1   1 1 1
0 0 0   0 0 0   0 1 0   1 1 0   1 1 0   1 1 0   1 1 0
```

This successive pattern tends to hide dot growth (dot gain) caused by ink soaking into the paper, smearing from paper movement, or distortion in old style display devices. The output will be somewhat grainy. (Note: this example, along with the others in this section, are for example only. They are not proven patterns).

With newer equipment, we don't need to compensate for dot growth, and so a dispersed pattern works much better. It still uses a successive patterning, but the new members are not clustered with the old. The resulting image is less grainy that with a clustered pattern. It my however produce artifacts.

```
0 0 0   0 1 0   0 1 0   0 1 0   0 1 0   0 1 1   0 1 1
0 0 0   0 0 0   1 0 0   1 0 1   1 0 1   1 0 1   1 0 1
0 0 1   0 0 1   0 0 1   0 0 1   1 0 1   1 0 1   1 1 1
```

The Bayer ordered dither is in very common use and is easily identified by the cross-hatch pattern artifacts it produces in the resulting display. This artifact creation problem is the major drawback of an otherwise powerful and very fast technique.

Error diffusion is the next step up and may remind you of the previous discussion on filtering. Once again I will quote, this time from DHALF.TXT, a compilation of information by Crocker, Boulay, Morra:

> "Error diffusion is very simple to describe. For each point in our image, we first find the closest intensity (or color) available. We then calculate the difference between the image value at that point and that nearest available intensity/color: this difference is our error value. Now we divide up the error value and distribute it to some of the neighboring image areas which we have not visited (or processed) yet. When we get to these later dots, we add in the portions of error values which were distributed there from the preceding dots, and clip the cumulative value to an allowed range if needed. This new, modified value now becomes the image value that we use for processing this point."

What this means is that the next step in dithering, error diffusion is an area process, not a point process as we have seen earlier.

Error diffusion generates the best results of any of the digital halftone methods so far. They produce images with less grain or noise then the previous methods discussed.

A short sample of diffusion algorithms, in what should be worst to best order:

Floyd-Stynberg
Jarvis-Judice-Ninke
Burkes
Stucki
Sierra

A 300dpi laser printer can approximate a 53 line screen. Most newspapers are printed with 85 line screens, and magazines are 120 or 133 lines. For example, PC Magazine currently uses 133 lines. Most image editing software will have several different halftone processes to chose from, depending on the printer and image to be printed. Your printer may have its own halftone options built in. Experimentation is necessary to find the best process for your system.

I will brake off this discussion here. Dithering should become less and less important as output devices improve.

Charge Coupled Devices

A charged coupled device (CCD) is a set of photosites, each produceing an electrical voltage in direct proportion to the amount of light that falls on it. This voltage can be used in its analog form, or converted to digital information using an analog to digital converter.

The photosites are normally arranged by one of several methods. First is the linear row, as found in scanners. Second is the two dimensional area array called an interline imager. The third is also a two dimensional area array called a frame transfer imager.

These devices do not require a shutter and are best suited to motion imaging.

A third style of two dimensional area array is the full frame imager. This device requires a shutter,and is best suited to still imaging.

Since the photosite is sensitive to all colors, the information gathered is gray-scale. Color CCDs are made with red, green, and blue filters installed on the device. The filter may be a simple stripe pattern, with one color filter covering a row of photosites. The filter may be a complex, and therefore more expensive, pattern. Some patterns may have 50% to 75% green, capitalising ont the fact the human eye in more sensitive to green. In this way more luminance information is gained with only a small sacrifice in color information.

Filters can cause information to be lost, as each color is separated by photosites of each of the other colors. Avoid this type of system if possible. Blue light falling on a red row is lost information, etc. Some programs may try to recreate this information through interpolation.

A better, but more expensive, system for color is the 3-CCD cameras in which the color is split by a prism and sent to its own CCD.

Color images can be also obtained from CCD devices designed for gray-scale use by capturing images with red, green, and blue filters, and then combining the images to form a color composite. Astronomical users have found the Wratten red #25, green #58, and blue #47 to work well. #23A light red, #56 light green, and #38A light blue are also used to keep exposure times down.

Several things need to be done to insure the best image quality if using a CCD device (most video cameras and scanners use CCDs). A Flat Field (white paper) image should be divided arithmetically with the image to compensate for differences in the photosites of the CCD. A Dark Field picture (lens cap on) can be subtracted from the output image to compensate for the thermal noise inherent in the CCD. (Note: this is more important for astronomical photographers who use long exposure times). Thermal noise decreases by a factor of 10 for every 32 degrees centigrade drop in temperature. Astronomical users (who use long exposure times and therefore collect more noise) will want to provide cooling for their CCD systems. CCDs with built in cooling are now on the market.

The CCD has the advantage over film of having a wider response range, and being more linear (very, very, linear) in response to light. The CCD provides a nearly exact record of the level of the received light.

If your CCD dosn't have an infrared filter built in you may wish to use an inferead filter, such as a Corion NR-400.

Color Separation

Color separation in used to convert information for the RGB model used by our monitor to the CMYK model used by most printers.

Most computer systems use RGB color. This is red, green, and blue.

This is an additive system. A computer monitor (or color TV) is a good example. The three colors are added to make other colors. All three equal makes a shade of gray. All colors full on makes white. All colors off makes black.

RGB color is different from subtracted color used by printers. In mixing paint, the three primary colors are yellow, red, and blue (sort of). In printing CMYK (cyan, magenta, yellow, black) colors are used. This is called subtractive (reflective) color. White light - red light = cyan (blue+green).

If you mix 100% C, M, and Y you should get black. Because print inks are not 100% saturated you end up with a muddy gray-brown. Also attempting to use 100% of each would be beyond the capabilities of most printers. An attempt to try this would leave a thick spot of ink, subject to smearing and other problems. And so we use K (K stand for black) and is used as a fourth channel to produce better print quality. Black ink is inexpensive, and easily used to adjust brightness. Generating this fourth channel is called *undercolor removal* or *gray-component replacement*. Here is one of the potential problems in color separation. What is the right percentage of K to remove? This number will vary printer to printer.

A good separation program should provide for *trapping*. This is a technique that prints the darker colors over the lighter colors to hide any misalignment during printing.

The angle of the print screen can cause *artifacts*. See the section on halftone for more information.

Something else to worry about is *dot gain* (*dot growth*). Ink soaking into the paper will spread.

Good loking color prints take lots of work.

Macintosh Files

If you have some hardware or software available on a Macintosh you can transfer image data back and forth to your PC via modem, network, or floppy disk using programs such as MacLinkPlus/PC. The current version 6.02 will only convert formats in 256 or less color or gray-scale, and you must do the conversion on a Macintosh system with color hardware.

Macintosh files, unlike IBM DOS files, may be stored in two separate files. These are the data fork and the resource fork. Most of the information useful to the IBM is in the data fork, but some file types may need information from the resource fork (font information is an example). Using the Apple File Exchange will result in only the data fork being transferred. Use Macsee (shareware) on the your IBM system to copy one or both. If you don't have Macsee, to transfer both forks, compress the file using Compact Pro or StuffIt. Decompress on the programs on the IBM side using the shareware programs ExtractorPC or UnStuffIt. This method will cause both data and resource forks to be transferred. You are on your own from here.

More Macintosh information: A MacBinary file will contain both data and resource forks. A Macintosh PICT file may contain both bitmaped and vector information. Use the TIFF format (as troublesome as it is) or make sure to write the information to the PICT file in "paint mode" (all data in bitmap format).

PICT/JPEG and QuickTime file formats use a lossy compression system.

Compression

Most new lossless compression programs use a Lempel-Ziv, Lempel-Ziv-Welch (LZW), or equivalent algorithm that preserves 100% of the data. They work fine on most image files and text files. Some file types, such as GIF, are already compressed. The basic idea in a lossless compression system is to look for repetitive data and replace it with a shorter code. 400 bytes of the same data can be replaced with one byte of the data and a command that says repeat the previous byte for a total of 400 bytes. Most programs will not compress the file if there is nothing to be gained by doing so. Normally there is no danger in trying, as these programs do not destroy a file when attempting to compress it. But you should always be safe and have a backup (archive) copy of important data, just in case.

An explanation of a losless compression follows:

Uncompresed Data	Compressed Data
It was the best of times, it was the worst of times, it was the age of wisdom, is was the age of foolishness	It was the best of times, @ worst @ @ @ age of wisdom @ @ foolishness

The compression program looks for repeated data. Repeated data ia replace with an token, represented here by the @, and a code where to find the data.

The first @ is followed by a 26 11. This says go back 26 and find a string 11 characters long. In this case @ 26 11 replaces "It was the". The second @ replaces "of times" with the token and 26 11, the directions about where to find the data.

As long as the token, the back count, and the string length flag are shorter than the original data, the file length is shortened. The trade off is the time needed to compress and decompress the file.

An older form of losless compression is Run Length Encoding (RLE). This method is simpler then other methods, but is capable of less compression then the newer methods. Example of RLE encoding follows.

Original data:

20 20 20 20 25 00 00 00 20 20
67 87 67 14 20 20 20 27 27 27
91 19 19 20 20 20 20 82 82 82

Data rewritten in repetition sequence:

20 20 20 20
25
00 00 00
20 20
67
87
67
14
20 20 20
27 27 27
91
19 19
20 20 20 20
82 82 82

Write a new file in the form:

Number of Repetitious	Data
04	20
01	25
03	00
02	20
01	67
01	78
01	67
01	14
03	20
03	27
01	91
02	19
04	20
03	82

The original file had 30 bytes, the compressed file has 14 bytes. This method works well with data containing bytes that repeat. Total random data, non-repetitive, can produce a file twice as big as the original!

Notice how LZW works on phrases, and RLE works on bytes. Some compression programs work on probable distribution.

Another form of losless encoding is Huffman. This method builds a "tree" that diagrams the data according to the distribution, and build your file from addresses that take you from the "Leaves" to the "root" (the data). Frano-Shannon is much the same but works from the root to the leaf. The Kodak Photo CD uses quantized Huffman encoding.

The two lossy compression systems you may see used are JPEG and Fractal. (There is a losless form of JPEG). I have little knowledge of Fractal.

JPEG compression (remember, it is a compression method standard, not a file format) uses several steps. First the image is converted to YUV (Y = luminance, U and V are chrominance). Redundant pixel values are then reduced using DCT (Discrete Cosine Transform). This is similar to Fourier transform (FT or FFT) but includes only the cosine part of the function. The DCT is applied to the image in an 8 x 8 pixel array. The resulting data is then quantized. RLE and Huffman encoding is applied to finish the compression.

Losless compression my give you from 20% to 100% file reduction with no loss of data. The amount of data compression is not controllable, and depend on the data itself. Lossy compression programs will normally give you control of loss vs file size. Depending on the output device 15:1 to 25:1 may be valid for "unnoticeable" loss. Only side by side compression, and knowing where to look, will show the loss. You must be the final judge.

The Future

THIS MAY BE THE MOST IMPORTANT SECTION IN THIS PAPER. THIS IS WHERE NEEDS, WANTS, AND NEW IDEAS COME TO LIGHT.

In written text the use of capitalized words, except where used for file or program names, is tantamount to SHOUTING. I use capitals here to get peoples attention in the hope that someone will clue me in to the fact that what I (we) are asking for is already available, or to clue the manufactures in that this NEEDS TO BE DONE.

First and foremost on the list of improvements to be made is in the area of input devices. Direct input is the only way to go. I have promoted the film camera, print, scanner method as a viable course of action during the writing of this paper. That it is, but it is not the best system.

We need HIGH RESOLUTION DIGITAL ELECTRONIC CAMERAS that can be attached to the PARALLEL PORT, or PCMCIA slot, of any computer. Forget adapted TV systems. Computer orientated equipment from start to finish! By using the parallel port, or a PCMCIA card, we NO LONGER NEED TO INSTALL A FRAME GRABBER. In this way, a LAPTOP OR NOTEBOOK system could be used in the field for direct image capture.

This camera should include a large CCD (at least 1024 x 1024), an exposure reading, ability to set the image size (V-pixels x H-pixels), and motor driven RGB filters. It should also use standard lenses and flash attachments. (Don't forget the cooled version for the star guys).

Because the "intelligence", and data storage, would be in the laptop computer, the price should be reasonable.

Another idea. If a camera system stored data on a PCMCIA card, we could swap in cards, just like loading fresh film. The data would be easy to transferred to any computer, including laptops, that has a PCMCIA slot. Why have proprietary cards, and readers, when the PCMCIA card is becoming a standard?

A process called Registration allows us to properly align combined (merge) images. Most of the programs I have tried do not allow you to register images by hand, let alone do it AUTOMATICALLY. And speaking of combining images, the best I have found so far can only combine two images at a time. We need to do four, better yet, EIGHT AT A TIME. An we need the ability to assign these to RGB channels while doing the combination. Below is a copy of what I requested of PhotoStyler.

Although PhotoStyler has many powerful tools, I find the ability to combine (merge, compute, etc) images to be far short of what I need in my application. I will state my requirements in the hope that you will consider incorporating these functions in the next release of PhotoStyler.

1. Combination of mutable images. Mutable, in this case maybe more than two. Four images is the absolute minimum, eight being a more useful number. For example, I may need to average several images together. If I were to do four images, I could do two and two, and then average the results. But what if I need to average an odd number of images, such as five?

2. Registration, the ability to align the images, is a must. This function should be controlled by the arrow keys, the mouse being much too course a tool for one pixel adjustments.

3. Image size should not be an issue. There should not be a requirement to have the images at a certain size, or in a certain size relationship to each other. Simply combine the images and the allow the user to trim off any overlap.

4. All calculations should be done with mathematical precision, no truncation allowed (except at the users request). Values above 255 and below 0 should be retained and the user prompted as to how to bring the image back into range (bias, histogram stretch\shrink, etc).

5. The above should work on images of ether 8 bit gray scale or 24 bit color.

6. In all of the above, if the images are 8 bit gray scale, they should be assignable to individual RGB color channels. That is, I need to bring in 3 (or more) gray scale images, assign them to individual RGB color channels, combine and register them, and produce a 24 bit color image.

7. A plus would be the ability to assign a luminosity channel, as a brightness control channel, at the same time. This image\channel would also need to be registered against the others during the combination.

Also needed is a set of ELECTRONIC CALIPERS. If we capture an image that contains a scale (measuring reference), we should have the ability to calibrate an electronic caliper against this reference, and use the electronic caliper to measure items in the image.

I hope that software companies will incorporate a TRACING PAPER FUNCTION in which a "tracing paper" could be placed over an image. A tracing could be made using the standard tools, and then this tracing could be lifted as a sperate image. (This can be done on some systems using the mask feature. See the section called Procedures for more info).

BETTER EXCHANGE BETWEEN IBM AND MACINTOSH systems, both in compatible file format and the diskette format is needed. PhotoShop for the Macintosh allows for file save as TIF IN IBM FORMAT. The Macintosh allows for the use of IBM formatted disks (using the DOS Mounter). IBM systems users, for the most part, have ignored the Macintosh (probably because of shame for the head start the Macintosh has in image manipulation).

Get the media (magazines, etc.) to give USEFUL RESOLUTION INFORMATION. Too many people are giving the number of pixels (photosites) as a total. For example "Image Sensor: 1/2-inch CCD, 200,000 pixels..." Is the 200,000 by 1? Or 100,000 by 2? Or... Come on, if needed we can multiply out the total by ourselves. So high by so wide is USEFUL information, a simple total is not.

While we are on the subject, why don't manufacture provide the color range their equipment can work with? No, not "24 bit color", this only tells us the number of steps the equipment can brake the colors into. Will to capture from ultra-violet through inferred? Or does it cut out some where in the blue range, leaving us with no purple\violet? Give this information in nm.

What if we were to capture an image containing the original, unweathered, rock (a rock broken open), the weathered surface, and two or more petroglyphs of different age? Could we then use the average values of the pixels of each figure to determine the relative age of each? (Example: compared to the current and original surfaces, figure 1 is 2% lighter than figure 2 and 5% darker than figure 3. Therefore, they are of different age?). Is this valid?

Garry Gillette opened a WHOLE NEW CAN OF WORMS at Rock Art '92. He started talking about stereoscopic imaging. We just got started talking, and then it was time to go. I am looking into the idea. Several schools of thought so far:

1. Stereoscopic (two images, one for each eye), "3D" (anagraphic?) red/blue displacement and the famous "3D" red/blue glasses. This can be done using a computer.

2. Laser holography.

3. Lenticular photos using a NIMSLO, Nishika, or other multi-lens camera.

4. Stereo photography using a Stereo Realist, Kodak Stereo, or other still camera.

5. Toshiba's 3D VHS Camcorder (experimental). Playback using LCD glasses.

6. Or the Stereo-CAD programs for the Atari 1040 computer using Tectronics LCD glasses.

How, if at all, any of this will tie into computers, and what we are trying to do, is unknown at this time.

Anyone interested in investigating 3D should download 3DMSG.ARC , STEREO.CO, and STEREO.THD from library 14 of the PHOTOFOURM on Compuserve. Or contact Me.

I would also like to have contact with JPL, NASA, Universities, and others who may be doing image enhancement using public domain software. I know that there are many people doing research in astronomy, satellite photos, art recovery, x-ray and medical imaging, and many other fields that may have useful information. Lets make sure the trickle down reaches all the way to us. Any leads? Please contact me if you do!

Software should make it VERY CLEAR, right up front, if the compression is LOSSY or NOT.
EXAMPLE:

Save File As

☐ Uncompresed

☐ Compressed (**Losless**)

☐ Compressed (**Lossy**)

See, it's not that hard to be user friendly.

Software should be able to SAVE LARGE FILES ACROSS MULTIPLE FLOPPIES. In a world which now supports 24 bit images, a floppy full error is inexcusable. Remember, this file splitting should be STANDARDIZED so that files are transferable between programs.

All color image editing programs must be able to do HSL (hue, saturation, and level) separation. Read this paper to find out why.

Coming soon? There may be holographic memory out by the end of 1994. 1 gigabyte of write once read many (WORM) storage in the size of a 5.25 inc drive. Removable media. Ten time faster than a hard disk. Can't wait!

HOT STUFF! Ed Scott has come up with an idea that may be of great value! His idea is to combine inferred images to regular images in order to enhance the contrast. This may be very useful, especially on petroglyphs on heavily patined rock!

Best regards

Terry

Technical Notes

This paper was created using WordPerfect 5.1. and DOS 6.0 on a '386\20mhz system.

Figures 1 and 2 were captured from Publishers Paintbrush using the Windows 3.0 Clipboard, and pasted back to Publishers Paintbrush. They were then stored as PCX files and imported to WordPerfect as PCX graphics.

Figures 5, 6, 10, 11, and 13, were created with Corel Draw 1.2 and exported as WordPerfect.WPG files. Figure 7 was generated in Corel Draw and exported as a PCX file. Note the different halftone process between it and figure 6.

Figure 6 and 7 are Lotus 1-2-3 version 2.2, graphs saved and used in Lotus .PIC format.

The scale (Appendix C) was created using Corel Draw, setting the Snap To Grid for centimeters, and then for inches. This sets the size of the squares to the required dimensions.

The images in Appendix A were scanned from a B+W photograph using a Macintosh system and saved as PICT files. The images were then converted to PCX format copied to an IBM formatted floppy disk using MacLinkPlus/PC. The image was edited/enhanced on the IBM using PhotoStyler and stored as PCX files.

Images for Appendix E, and figures 3 and 4, were created using CA Cricket Paint as grayscale TIF files. Wordperfect was instructed to use them as black and white images.

Figure 12 was scanned from a photocopied page of a magazine, using a neighbors Caere Typist Plus Graphics hand scanner and Caere Graphic Editor software. The file was saved as a grayscale TIFF image on a floppy disk. Using mt system, I opened a new file in CA-Cricket Paint, the same size as the scan, and then brought the scanned image in as a mask. The masking was turned off, but the show mask was left active. In this manner I could see the image as a mask, but could "paint through" to the "canvas below". I then traced the image using the line and curve tools. Curved areas were done in sections. A combination of CA-Cricket and PhotoStyler tools were used to add the details, these done by hand. The image was saved, and used, as a PCX file.

This paper was printed on a 24 pin dot matrix printer at 180 dpi. It takes over 2 hours to print one full document (graphics, in the form of pictures and tables, are slow to print!). I use an external fan on the printer to keep the print head cool. The file size for this paper is over 855Kb (un-compressed).

To give credit where credit is due:
I use an American Systec Corp. '386 20Mhz system with 8Mb of memory, IIT co-processor, and a Microsoft mouse. Video is via a Western Digital Paradise 1024 VGA (512Kb memory) and a Ceptre monitor (for a 256 color 640 x 480 pixel video system). My printer is an Epson LQ-510 (dot matrix, 24 pin). I run (as a base, I use many other things as needed) DOS 6.0, Windows 3.1 (when needed), PC-Kwik Power Pak 3.0, QEMM386 6.01, WordPerfect 5.1, and the

Norton Utilities 6.01. This combination has proved to be a stable and useful base to work from.

Appendix A Pictures

Top Left: Original.
Top Right: Negative.
Bottom Left: Sharpen.
Bottom Right: Brightness and Contrast.

Appendix B

This is a partial listing of commercial programs and some useful tools. Does not include shareware programs mentioned in the text. ? = DOS or Windows requirements unknown.

Top Price (over $500) Image Editing Programs:

PhotoStyler (requires Windows)
Aldus Corp. (purchased from U-Lead System, very good)
(get the free demo from your local BBS)

PhotoShop for Windows (perhaps the best overall image editing
Adobe Corp. program)

Digital Darkroom (?)
Aldus Corp.

PicturePro (Windows)
Ventura Software

Image-In-Color (requires Windows)
Image-In Inc.

Image-In-Color Professional (requires Windows)
Image-In Inc. (good, get the free demo an try it out)

ImagEdit (requires Windows)
IBM Corp. (gray-scale only)

Picture Publisher (requires Windows)
Micrografx Corp.

Tempra Pro (?)
Mathematica Inc.

PixFoto (DOS)
PixoArt Corp.

Mid-price ($250 to $500) Image Editing Programs:

Publishers Paintbrush (requires Windows) NO LONGER SUPPORTED
Z-Soft Corp.

CA-Cricket (requires Windows) (no bit level adjustments) (looks

Computer Associates International Inc. good at first glance)

WinRix (requires Windows)
RIX Softworks (first release full of bugs)

Gray F/X (DOS)
Xerox Imaging System Inc. (gray-scale only)

Image-In Plus (?)
Image-In Inc. (gray-scale only)

Lo-price (up to $250) Image Editing Programs:

PC Paintbrush IV Plus (DOS)
Z-Soft Corp. (mostly a paint program with scanner control)

PhotoFinish (requires Windows)
Z-Soft Corp. (looks very promising from the preliminary info)
(get their free demo)

ImagePrep (requires Windows) (not bad)
Computer Presentations

PHIPS (DOS)
Computer Associates International Inc.

Zip Image (DOS)
Catenary Systems (gray-scale only)

Tempra GIF (?)
Mathematica Inc.

Apple Macintosh to IBM PC Transfer Program:

MacLinkPlus/PC (DOS and\or Windows)
DataViz Inc. (does contain translators, with restrictions, in my opinion this program is junk when it comes to translation image files!)

Mac-in-DOS (DOS and ?)
Pacific Microelectronics (does not, at this time, contain translators)

Image Compression Programs:

Picture Packer (DOS or Windows versions) (lossy JPEG compression)
Video & Image Compression Corp.

Images Incorporated (Windows, fractal compression)
Iterated Systems, Inc.

File Format Conversion and\or Screen Capture:

Note: Some of these programs contain fair to very good imaging processing tools.

Picture Eze (requires Windows)
Applications Techniques Inc. (many useful file formats)

Envision It (DOS)
Envisions Solutions Technology Inc (bitmap to vector)

HiJaak (DOS or Windows versions)
Inset Systems Inc.

Halo Desktop Imager (requires Windows)
Media Cybermetics (good)

Conversion Artist
North Coast Software Inc. (requires Windows)

DoDot (requires Windows)
Halcyon Software Inc.

Image Pals
U-Lead Systems

Scanner/Monitor/Video Card/Printer Test and Adjustment:

FIPS 157 Guideline for Quality Control of Image Scanners. $62
National Technical Information Service
Document Processing Branch
5282 Port Royal Road
Springfield VA 22161

Q-60C Color Reflection Test Target
Kodak

ANSIIT8 Technical Subcommittee, Working Group 11
Information available from National Printing Equipment and Supply Association, Reston, VA

DisplayMate (DOS) (top-of-the-line monitor test/calibrate software)

Sonera Technologies

Image Cataloging Programs:

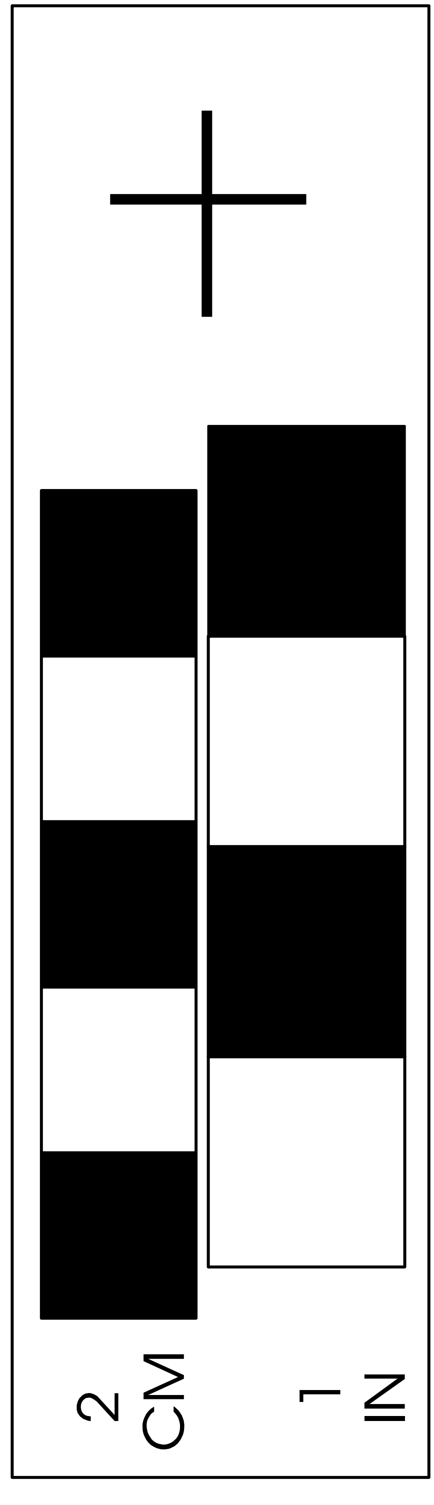
PubTech Multitrack (?)
Publishing Technologies Inc.

Scrapbook Plus (?)
Eikon Systems

Image Presentation:

ImageQ (requires Windows)
Image North Technologies

Appendix C



Appendix D

Image File Types

Format	Ext.	Description
Encapsulated PostScript (EPS)	.EPI EPSI	Subset of PostScript. Gray-scale, RGB, and indexed images.
GEM VDI Image	.IMG	1, 4, 8 bit. See note 2
Graphics Interchange Format	.GIF	Compressed 1 through 8 bit images, Interleaved and non-interleaved. Versions: GIF87A and GIF89A (text added, preserves aspect ratio). Good compression (LZW).
Hewlett-Pakard Graphics Language	.PCL	Used by HP and compatible printers. Can include text and vector information with the bitmap data. 1 Bit only.
Hewlett-Pakard Raster Transfer Language	.RTL	Used for HP color printers. Compressed and un-compressed.
Joint Photographers Experts Group See note 3	.JPG JPEG	Lossy (adjustable for file size vs data loss) compression. Several incompatible formats. JPEG, JFIF AND TIFF\JPEG formats. May be interleaved
Interchange File Format	.IFF .CE	Amiga Computers 1 through 8 bit and 24 bit. May include images, text or music.
InterLeaved BitNap	.IBM ILBM	Amiga Computers 1 through 8, and 24 bit images.
Macintosh PICT/PICT2	.PIC PICT	Widely used Apple Macintosh format. 1 bit to 32 bits. Many options may cause problems.
PCX	.PCX	Widely used format. 1,4,8, and 24 bit with minor compression (RLE).
DCX	.DCX	1 bit version of PXC for FAX use.
Tagged Interchange File Format	.TIF TIFF	Widespread 1, 4, 8, 12, 24, and 32 bit with several types of compression. Complexity has created many incompatible files. Apple Macintosh TIFF has slightly different header.
Targa	.TGA	Compressed and un-compressed 8, 15, 16, 24, and 32 bit formats.

Vivid	.IMG	Vivid Ray Tracer format. See note 2
Windows and OS/2 Bitmap	.BMP .DIB .RLE	Compressed and un-compressed 1, 4, 8, and 24 bit format. DIB is un-compressed. RLE is compressed.
WordPerfect Graphics File	.WPG	1, 4, or 8 bits with compression. May contain text and vector information. Version 5.0 and 5.1 differ.
Pictor PIC	.PIC	1 or 8 bit. Not compatible with other types of PIC files.
PC-Paint PIC	.PIC	4 Bit. Not compatible with other PIC files.
Dr. Halo	.CUT	2 or 8 bit. Palette information in separate PAL file.

Note 1: Four digit extensions are for Macintosh or other computer systems.

Note 2: GEM and Vivid both use .IMG extensions.

Note 3: The JPEG standard doesn't set forth a format, only the compression method. All ".JPG" files are based on some other file format, or proprietary format, and compressed with JPEG. There will be some problems with comparability.

Appendix E

Linear histogram\gamma adjustments work well with an image which has narrow information range, such as a low contrast image. Non-linear adjustments work well with images that have a wide contrast with a heavy concentration of data in one area.

References

PC Magazine, various articles over the last several years.

PC Sources, various articles over the last several years.

PC World, various articles over the last several years.

Byte, various articles over the last several years.

Sky and Telescope, various articles over the last several years.

Computer Images, Time Life Books.

Color and the Computer, edited by H. John Durrett.

Datamation, April 15, 1990, p96.

The Winn Rosch Hardware Bible, Brady, 1989.

High Color Magazine

Advanced Imaging Magazine *

New Media Magazine *

Color. Why the World Isn't Grey, Hazel Rossotti, Princeton University Press, 1985.

Color. Origin, systems, use. Harald Kupper, Van Nostrad Reinhold Ltd, 1972.

The Reconfigured Eye. Visual Truth in the Post-Photographic Era, William J. Mitchell, The MIT Press, 1992.

Physics Today, December 1992.

* Trade Magazines

Documentation and help files from many different programs!

Information from several BBS Forums including The Photofourm, Dtpforum, and Graphsup forums on Compuserve.

Index

'286 6-6, 8-7, 9-1, 9-2, 9-3
'386 6-6, 6-12, 9-3, 12-1
'486 6-6, 9-3
15 bit 6-11
16 bit 6-1, 6-11
24 bit 5-6, 5-7, 6-1, 6-3, 6-5, 6-10, 6-11, 6-12, 6-17, 8-2, 8-6, 9-2, 9-3, 10-12, D-1, D-2
3-CCD 10-15
32 bit 6-1, 6-11, D-2
32 bits D-1
35mm 6-3, 6-11, 6-15
4mm 6-14
4mm Tape 6-14
8 bit 5-7, 6-1, 6-5, 6-9, 8-2, 9-1, 9-2, D-1, D-2
80286 9-3
80386 9-3
80486 9-3
8088 9-1, 9-3
8514/A 6-10
8mm 6-14
8mm Tape 6-14
Achromatic 4-1
Anti-aliasing 6-11, 6-12
Bernoulli Box 6-14
Bitmap 5-2, 6-15, 10-17, B-3, D-1
Bitmaped 10-17
Bitmapped 5-1
Calibration 6-11, 7-1, 8-2
CCD 6-1, 6-2, 6-5, 6-16, 7-6, 10-11, 10-14, 10-15, 11-1
CCITT 6-19
CD-ROM 6-4
CGA 6-9, 6-10, 10-12
Chromatic 4-1
CIE 5-3, 7-9
Clipboard 5-2, 7-3, 7-6, 12-1
CMYK 5-2, 6-5, 6-16, 7-1, 7-3, 8-4, 10-16
Co-processor 6-6, 8-1, 9-3, 12-1
Color CCD 10-15
Color separation 6-16, 7-5, 10-16
Compression 6-11, 6-14, 8-4, 8-5, 8-6, 9-2, 10-17, B-2, D-1, D-2
Contrast 4-1, 4-2, 7-1, 7-3, 7-6, 7-7, 8-2, 10-3, A-1
Conversion 5-7, 6-3, 7-2, 8-1, 8-3, 8-6, 8-7, 9-2, 10-16, B-2, B-3
Convert 5-7, 6-15, 8-3, 9-2, 10-16
Converting 4-2, 5-7, 10-10

Convolution 10-1, 10-10
Corel Draw 6-11, 12-1
Dither 8-2, 10-14
Dithered 10-12
Dithering 10-12, 10-13, 10-14
DOS 6-1, 6-12, 8-1, 8-6, 8-7, 9-1, 9-2, 9-3, 10-16, 11-3, 12-1, B-1, B-2, B-3
Dot gain 10-16
Dot growth 10-16
EDSI 6-13
EGA 6-9, 6-11, 10-12
EISA 6-1
Electronic still camera 6-1, 11-1
Electronic still cameras 6-1, 6-2
False color 4-2, 9-1
Figures 5-2, 5-6, 6-7, 6-8, 6-20
File format 5-6, 5-7, 6-15, 6-16, 8-1, 8-3, 8-6, 8-7, 9-2, 11-2, B-2, B-3, D-1, D-2
Film 6-3, 6-11, 6-15, 7-1, 7-2, 9-4, 10-15, 11-1
Filter 7-3, 8-1, 10-1, 10-2, 10-4, 10-10
Filters 4-2, 6-5, 7-2, 7-5, 8-2, 8-3, 10-1, 10-3, 10-4, 10-5, 10-7, 10-8, 10-9, 10-10, 10-15
Floptical 6-14
Floptical Disk 6-14
Fractal 8-5, 10-1, B-2
Frame capture 6-2
Frame grabber 6-1, 6-2, 6-3, 6-4, 11-1
Frame process 10-1
Gamma 7-2, 8-2
Geometric process 10-1
Gray-component replacement 10-16
Halftone 6-5, 6-17, 7-2, 10-1, 10-11, 10-12, 10-14, 10-16
Halftone mode 6-5
Halftones 8-2
Hercules 6-10
High Color 6-11
HIS 7-5
Histogram 4-1, 7-6, 8-1, 8-3, 10-9
HLS 7-5
HSV 5-3, 7-3, 7-5, 8-4, 10-10
Huffman 6-3, 10-18, 10-19
IDE 6-13
Indexed 5-7, 6-5, 6-9, 8-2, 9-2, D-1
Interlaced 6-2, 6-10, 6-11, 6-16
Interpolated 7-6
Interpolation 6-5, 10-15

ISA 6-1
JPEG 8-4, 8-5, 9-2, B-2, D-1
Kodak 6-3, 6-4, 10-19, 11-3
Laser Disk 6-14
Lines 6-2, 6-3, 6-11, 6-15, 6-17, 7-2, 7-7, 10-12, 10-14
Lotus 12-1
Lotus 1-2-3 5-7
LZW 10-17
Macintosh 2-1, 5-6, 8-7, 9-1, 9-2, 10-16, 10-17, 11-2, 11-3, 12-1, B-2, D-1, D-2
Mask 7-5, 7-6, 7-7, 12-1
Masking 7-5
MCA 6-1
MDA 6-9
Merge 4-2, 7-5, 8-2, 10-10, 11-1
Merged 6-2
MFM 6-13
MGA 6-9
Modem 6-18, 7-4, 8-4, 8-6, 9-2, 10-16
NTSC 6-11, 6-15
Object-Oriented 5-1
Opto-magnetic 6-14
OS/2 6-6, 6-12, 9-3, D-2
Pantone 7-1
PCMCIA 11-1
Photo CD 6-3, 6-4, 10-19
Photosite 5-3, 6-5
Pixel 4-3, 6-10, 6-11, 8-2, 10-1, 10-2, 10-3, 10-4, 10-5, 10-8, 10-9, 10-10, 12-1
Pixelation 7-6
Point process 10-1
Print 6-3, 6-4, 6-11, 6-15, 6-16, 6-17, 7-1, 9-1
Printed 10-11
Printer 5-1, 5-2, 6-16, 6-17, 7-1, 7-2, 7-6, 9-4, 10-11, 10-12, 10-14, 10-16, 12-1, B-3
Printers 6-1, 6-5, 6-16, 6-17, 6-18, 7-2, 10-11, 10-12, 10-16, D-1
Printing 5-2, 6-16, 7-6, 8-1, 8-4, 10-16
Prints 6-2, 6-3, 6-15, 6-16
Raster 5-1, 5-7, 6-15, 8-1, 8-3, D-1
Red, green, and blue 5-2, 6-5, 6-11, 10-15, 10-16
RGB 4-2, 5-2, 6-5, 7-5, 8-4, 10-2, 10-10, 10-16, D-1
RLE 5-7, 10-17, 10-18, 10-19, D-1
RLL 6-13
Scanner 5-1, 6-4, 6-5, 6-12, 7-2, 8-1, 8-2, 9-3, 9-4, 11-1, B-2
Screen frequency 6-17
SCSI 6-13

Shareware 5-7, 7-1, 8-5, 8-6, 8-7, 9-1, 9-2, 10-17, 12-1
Sharpen 4-2, 7-3, 8-3, 10-2, 10-4, A-1
Sharpening 7-5, 10-4, 10-10
Slides 6-3, 6-15
SVGA 6-9, 6-10
TIFF 6.0 5-6
Trapping 10-16
True Color 5-1, 6-3, 6-10, 6-11, 6-12, 6-13, 6-18, 8-2
TSR 8-6
TV 6-1, 6-2, 6-4, 6-15, 10-13, 11-1
Undercolor removal 10-16
Vector 5-1, 5-7, 6-15, 8-1, 8-3, 10-17, D-1, D-2
VGA 6-9, 6-11, 6-15, 8-2, 10-12, 12-1
Video camera 5-1, 6-2, 6-3
Video capture 6-2, 8-4
Video tape 6-2
Windows 5-7, 6-1, 6-6, 6-12, 6-13, 8-1, 8-6, 9-1, 9-2, 9-3, 12-1, B-1, B-2, B-3, D-2
WordPerfect 1-1, 6-11, 12-1
WORM 6-14, 11-4
XGA 6-10
XT 9-3