

Designing a Website: Graphic File Formats on the Web

**Virginia R. Hetrick, Ph. D.
(drjuice@gte.net)
310.471.1766**

and

**Donald McClelland
(donm@fmch.ucla.edu)
310.825.4659**

**Fowler Museum of Cultural History
University of California Los Angeles
Los Angeles, California 90095**

**Southern California OS/2 User Group
1998 Open House**

The material in this document represents the combined thoughts of Don McClelland and Virginia Hetrick on a number of issues related to manipulating images and displaying them across the Web. We put these together in the context of a project we have been working on which is called the Museum Educational Site Licensing Project. The opinions expressed within are ours and do not represent any official position of the Fowler Museum of Cultural History, The Regents of the University of California, or the Museum Educational Site Licensing Project.

Before we start, please note that throughout the various sections, our comments are made in the context of the kinds of images most likely to be used in the MESL project. So, for example, we will not consider images which are only black/white such as might be created by scanning text for optical character recognition. The MESL project is designed to explore the issues of licensing works of art, photography, and cultural artifacts from museums of many kinds by educational institutions at the university level by prototyping site licenses for a small number of universities. Each participating museum contributed images of about 2000 objects to the project. Faculty from regional studies programs, photography, art history, and anthropology (among many others) have participated in the project by designing courses which use the object content as the focus of lectures and student research projects.

We have noticed that there is a great variability in the backgrounds of the MESL project's participants and this is one attempt to start those people on a more level playing field. We believe that this information will also be useful to SHARE member installations as they begin developing their own Web sites.

How This Material Is Organized

To reduce the mindbending load that folks could get from our long thoughts responding to specific questions that prompted this effort, we have split this into a number of parts.

Don McClelland, the Fowler Museum team leader and the Fowler's Director of Information Systems, and Virginia Hetrick, the Fowler Museum network lady and UCLA Institute of Archaeology Scientific Computing Manager at the time this was written, have put these thoughts together. In several places, there are tables (mostly Don's work) and lists (mostly Virginia's work) which were constructed using spaces rather than tabs. If you have an electronic copy of this document, to be certain these tables print correctly, you should use a monospaced font at 10 points (equivalent to 12 pitch), such as Courier New in DOS/Windows environments, Courier in Macintosh environments, or the system font in Unix, plain DOS, or OS/2 environments.

How to Approach This Topic

The second part contains some definitions which we will use in the third part. It also attempts to give some image processing background to people who may feel uncomfortable with the fairly technical discussion in the third part. If you think you are

an image nerd, you might want to skip reading the second part in detail just now and get right into the third and fourth parts.

Next, for those people who do not consider themselves to be image nerds, we would like to preface the hairy technical parts with a couple of points:

- ◆ What we think you should be concerned with is what looks good to you. If you are not an image nerd and are involved in this project, you probably have a pretty good "graphic eye" and can tell when something is a reasonably good reproduction of the original and when it is not.
- ◆ If the gory details of what we were talking about in the last two parts are mysterious, try taking a look at the second part. Alternatively, you might just want to leave these mysterious details as mysteries. I (Virginia) absolutely believe there are certain things about the museum and instructional aspects of this project that I would just as soon leave as mysteries for me and let somebody who has a different background handle them. ;-)
- ◆ Finally, if there are things you would rather not leave as mysteries and we have not been clear about explaining them, please let us know and we will try to clear things up for you.

Definitions

This part consists of a glossary of digital image processing terms, mainly only the ones that people who lack a heavy-duty image processing background may need. The objective is useful information, not formal definitions. Consequently, the terms are ordered in the glossary in the order that we think it makes sense to learn the terms, i.e., not alphabetically. This part was written mainly by Virginia with input from Don.

Byte: the most commonly referenced unit by which digital information is stored. When looking at text materials, generally one character is represented by one byte. A byte consists of eight bits. Each bit can take the value of 0 (off) or 1 (on). All of the possible variations of zeroes and ones allow a maximum of 256 different combinations within a single byte. A kilobyte is 1024 bytes, a megabyte is 1,048,576 bytes (1000 kilobytes), and a gigabyte is 1,063,741,824 bytes (1000 megabytes). These are frequently abbreviated as KB, MB, or GB, respectively. Additionally, 1 kilobyte is sometimes called 1K while 1 megabyte is often called 1 meg.

Pixel: shortened form of the words "picture element", sometimes also called pels (on the right side of the Eastern pond). A pixel is the smallest resolvable unit in a digital image. A particular object cannot be identified from a single pixel because objects have shape and/or dimension that require more than a single pixel to represent them. The most usual representations of a pixel in color images are either as 8-bits (1 byte per pixel) or as

24-bits (3 bytes per pixel) [see the discussion of color resolution below]. By custom, pixels are considered to be square today simply because dealing with non-square pixels becomes a nearly intractable problem very quickly.

Scanline: in a digital image, pixels are arranged in rows called scanlines. Documentation of an image includes the number of pixels on one scanline and the number of scanlines which comprise that image.

Digital image: an image that has been captured in some way by a sensor which converts the captured information into a series of numerical values. These numerical values generally range between 0 and 255 or 1 and 256, although they can take on other values if they are defined to do so and the sensor can convert the information it captures into some other range. Generally, when the spatial size of an image is discussed, the number of pixels in one horizontal row is given first and the number of scanlines presented vertically is given second. So, for example, saying that an image is 640x480 means that there are 640 pixels in one scanline and 480 scanlines in the image.

Non-coded information: digital images represent a class of information called "non-coded" information. The simplest definition of "non-coded" information is that it is only possible to evaluate a particular part of the information when reference is made to the whole. Waveforms that represent sound are another form of non-coded data. This is important in the context of the MESL project because it explains why we are concerned with how a particular digital representation of an image either holds up or deteriorates with repeated instances of compression and decompression. If we do not have reference in the stored image to the entire set of information contained in the original image, usually we cannot exactly reconstruct a real representation of the original image.

Resolution: the term "resolution" can apply to two different pieces of information in the present context. The first is spatial resolution and the second is color resolution.

Spatial resolution refers to the fact that any given image can be digitally represented in several ways. For example, I might have a picture postcard. If I scanned the postcard in using my HP ScanJet IIC, I could specify that I wanted the picture scanned at 75 dots per inch (dpi) or 300 dpi or some other resolution that I might choose. The specification of 75 dpi is generally considered to be display resolution while 300 dpi is generally considered to be print resolution. To see how these are different, try marking the edge of a piece of paper at one-inch intervals. Then, place the piece of paper against your computer screen with two of the tick marks on points about 1 inch apart on the screen. Then, start zooming in on the picture until you can see distinct pixels. At this point, you should be able to count the pixels between the two points you started with and see about 75 dots between your original points. If you wanted to print the picture and had captured the picture for that purpose, your two points, when examined under a magnifying glass or by software zooming, would be found to be 300 dots apart. In the remainder of these parts, spatial resolution can also be defined by some qualitative terms that have a quantitative implication, especially when discussing Kodak PhotoCD

technology. The exact details will be described when we Don discusses various image technologies in Part III.

Color resolution refers to the fact that it is possible to see color in several different ways. If the physiological aspects of vision are considered, most men can see something less than one million colors (many times it is much less, particularly if the man is colorblind). On the other hand, women can typically differentiate something like 1.5 million distinct colors and specially trained women in the textile industry have been found to be able to differentiate as many as three to four million colors. In practical terms for the MESL project, this means that we have more than a little room to wander and experiment in. If you think about it, an image which measures 640x480 can have a maximum of 307,200 colors in it because any given pixel can be represented at one time by only one color, regardless of how that color is defined. By contrast, an image that measures 1K on a side could have, at most, 1,048,576 colors, still only 1/16th of the maximum number of colors which can be represented in a 24-bit image.

The other important aspect of color resolution is that a color is usually represented in one of two ways, digitally speaking. The first is called 8-bit color and means that a single byte represents a particular color. This is usually the color resolution of images found on most microprocessor-based systems such as PCs and Macs. Which portion of the 8 bits is used to represent red, green, and blue depends on the particular characteristics of the monitor and its associated graphic accelerator (display) card.

At most, the number of colors it is possible to represent in such a picture is 256. The second way to represent color is called either "true color" or 24-bit color. This representation is called "true color" because the "actual" color of an object can be accurately represented, even for the most discerning eye, using whichever of 256x256x256 colors (16,777,216) is closest. It is possible to represent each of the three additive color primaries (red, green, and blue) by devoting 8 bits of information to each primary. This representation is also called 24-bit color because there are 24 bits in the 3 bytes of information needed to represent this color model.

Color models: Two color models will be mentioned in the course of our discussion. RGB stands for Red-Green-Blue which is the most commonly used 24-bit color model. In the RGB model, one byte represents the amount of red, another byte represents the amount of green, and a third byte represents the amount of blue. If all three colors are at their full saturation, the result is a white pixel. This is why this color model is referred to as additive (if you add all the colors together you get white). If all three colors are at their minimum saturation, the result is a black pixel. A "full" yellow pixel is created when both red and green are at maximum saturation and blue is at its minimum saturation. When a "full" purple pixel is created, both red and blue are at maximum saturation and green is at its minimum saturation.

When we begin talking about PhotoCDs, you will hear that the color model for PhotoCDs is YCC, which represents one byte for luminance and two bytes for two

different aspects of chrominance (called C1 and C2). An equation exists for converting YCC to the more commonly used RGB color model.

Dithering: to dither is to use colors which are not the one being represented. These are presented in combinations to emulate the represented color to the person viewing the image. A number of "official" dithering schemes that can be implemented in software are available. The most popular are Floyd-Steinberg and ordered dithering. Nearly all software products today automatically produce dithering when they cannot accurately represent colors. Thus, in general, it is not terribly necessary or useful to worry about the various dithering schemes. The most obvious way to see what dithering is all about is to use Microsoft (circle R) Windows in a VGA system. Since Windows on a VGA system only has access to 16 colors, it must necessarily provide dithering to emulate the colors it cannot represent properly. How such emulation works is to represent a pixel by some number of pixel positions on the screen, typically four, nine, or sixteen, so that the characteristic square shape of the original pixel is retained. A very short example might be to represent a pixel that is 0-0-.5 in a 24-bit image (this means it has zero red, zero green, and half blue intensities in the 24-bit image). To represent this same pixel in a dithered image on an 8-bit display, four pixel positions on the display could be used. Two of the pixel positions (usually the opposite corners) would be black (0-0-0) and two of the pixel positions could be full blue. An alternative representation would be to represent all four pixels as 0-0-.5 keeping in mind that the range of blue on an 8-bit system is only about 1/32nd or 1/64th of the possible range of blue on a 24-bit system. Generally, the first of these alternatives, using black with full blue for dithering, will give a better looking result.

Lossy and lossless images: Lossy and lossless refer to the relative amounts of information present in a stored image and give an indication of the degree of certainty we can have about being able to reconstruct an original image from its stored representation. If an image is stored by a lossless method, an exact reconstruction is possible. If an image is stored by a lossy method, an exact reconstruction is not possible, but a reasonably complete reconstruction may well be possible. Lossy methods generally give the person storing the image some degree of choice (either quantitative or qualitative) in determining how much information may be lost in the storage process.

Compression: many of the various mechanisms used to store digital information (not just images) apply some form of compression in order to minimize the amount of storage space required to save the information. Compression depends on there being redundancy in the information being compressed. To compress an image means to represent it in a different digital format and in such a way that the image can be decompressed without looking different from the original image used to create it. Different compression methods affect image content in different ways. Usually, when one of these methods is used, the last element in the file name of the image (the file extension in a PC environment) indicates the compression method or storage format. All of the compression methods outlined below are industry standards with the exception of

JPEG and MPEG. These two began as industry standards and are in the process of being adopted as national and international standards.

Without going into the gory details, three of the most common image storage formats using compression are described below.

- ◆ JPEG or JPG - pronounced "jay-peg", the abbreviation stands for Joint Photographic Experts Group. The most flexible of a number JPEG formats is referred to as JFIF which stands for JPEG File Interchange Format. Non-JFIF JPEG files are usually proprietary to a particular vendor and should be avoided. JPEG incorporates the DCT (discrete cosine transform) to compress images.
- ◆ GIF - pronounced like the peanut butter, "JIF", GIF is an abbreviation for Graphic Interchange Format, a proprietary format developed by Compuserve for the use of its subscribers. It was initially developed because color images were becoming more popular and none of the image storage formats available at the time GIF was developed could handle color images. Those other formats do now handle color images without difficulty. However, GIF remains popular. GIF incorporates a compression algorithm (LZW) licensed to Compuserve by Univac; Univac began in the summer of 1994 to require enforcement of the patent on its compression algorithm and it is not clear how this will shake out or what its impact on the ubiquity of GIF images will be.
- ◆ TIFF or TIF - Tagged Image File Format, a proprietary format developed by Aldus and which is commonly found in microprocessor paint software packages. TIFF images can be compressed or uncompressed, if compressed, they are usually compressed using LZW compression though the TIFF 6.0 standard also permits DCT compression.
- ◆ Kodak Photo CD (PCD files) - Photo CD details are discussed at length in Part III. The PCD specifications are held by Kodak as proprietary information, although Kodak has released software development kits for Windows, Macintosh, and Unix which allow programmers to develop applications which incorporate the PCD formats. PCD images are stored using PhotoYCC format (also proprietary). PhotoYCC can be readily converted into other forms which are readable by a large number of software programs. If you decide that you are going to use PhotoCD images, be sure that your CD-ROM drive is capable of reading multi-session PhotoCDs (this is usually noted on the packaging of the CD-ROM drive). PhotoCD images come in six flavors, of which we will be interested in five. Those five are Base/16 (read as Base over 16), Base/4 (Base over 4), Base, Base*4 (Base times 4), and Base*16 (Base times 16).

Additionally, several other terms related to storing images may pop up when we are discussing image quality and formats:

- ◆ LZ or LZW - Lempel-Ziv or Lempel-Ziv-Welch compression is most commonly used in conjunction by other formats such as GIF and TIFF rather than being used alone (LZW is proprietary to Unisys)
- ◆ RLE - Run-Length Encoding is used by TIFF (early versions), TGA (see below), BMP (used to store Windows and OS/2 system bitmaps), and PCX (Z-Soft's *PaintBrush* image storage format). It is nearly always only a compression method and not a storage format itself.
- ◆ TGA - Targa format, a proprietary format developed by AT&T before it spun off the TrueVision Company to employees. Targa images, if they are compressed, use RLE. However, most often, they are not compressed and are therefore HUGE.

Finally, you may hear about MPEG (pronounced em-peg) or MPEG-2. MPEG is an acronym for the Motion Picture Experts Group which has defined an industry standard for the digital storage of multiple frame imagery. Generally, these images will be either motion pictures, full-motion video, or animation sequences. Though it is possible to store single frame images using MPEG, it is unusual for them to be stored this way.

You can "Read All about It" in the following two references:

Murray, James D., and William vanRyper. (1994) *Encyclopedia of Graphics File Formats*. Sebastopol, CA: O'Reilly and Associates. ISBN: 1-56592-058-9 Price when I bought this book: \$59.95

Pennebaker, William B., and Joan L. Mitchell. (1993) *JPEG Still Image Data Compression Standard*. New York: Van Nostrand Reinhold ISBN: 0-442-01272-1 Price when I bought this book: \$59.95

Considerations while selecting image resolutions

Please be certain you have understand Part I before you read this part. This part was mainly written by Don with input from Virginia.

This document summarizes some of the considerations in selecting image resolutions for the MESL project. The document includes two tables. To assure that they will display and print correctly from all email systems, I have aligned the columns with space characters. Please select a monospaced font such as Courier when displaying or printing this document so that the columns will line up correctly.

Digital images to be used in the MESL Project, as in other digital imaging applications, must give their users visual information with sufficient detail and quality to be useful and informative; however, the images must be storable in a reasonable amount of disk space and must be quickly and conveniently accessible to their users. To meet these conflicting objectives, we must give careful attention to the image specifications; especially, image size, color resolution, and compression method.

Image Size

Throughout this discussion, keep in mind that we are primarily interested in two "sizes: (1) the amount of random access memory (RAM) the image occupies when it is opened for viewing and (2) the amount of space it occupies when it is stored on the disk. The image can be stored on the disk in either a compressed or an uncompressed format. When it is stored in an uncompressed format, the space it occupies on the disk is approximately equal to the amount of RAM it occupies when it is open for viewing (its "uncompressed" or "expanded" size). If it is stored in compressed format, it can be much smaller (its "compressed" size).

The size of a digital image is determined by its spatial and color resolution (see definitions) and is selected partly by choice, based on practical considerations to be discussed later, and partly by limitations of the equipment used to create and display it.

We can make a digital image of an object by photographing it directly with a digital camera or with a video camera and digitizing board. Another way is to photograph the object with a conventional film camera, and then scan the print or transparency with a flatbed scanner or transparency scanner. The resolution of each of these devices is limited by its optics and/or its electronic imaging sensor. The resolution of a photograph is limited by the size and distribution of its silver or dye grains, which correspond to the pixels of a digital image, although they are distributed randomly instead of in neat rows and columns.

After we have created the digital image, we can view it by:

- ◆ Displaying it directly on the computer screen,
- ◆ Projecting it on a wall screen using a digital image projector,
- ◆ Printing it using a laser printer or other type of computer printer, or
- ◆ Converting it to a film transparency or negative using a film recorder.

Each of these devices has its own resolution limit. The resolution of the final displayed image will be no better than that of the lowest resolution device used to capture or display it. If we intend to display an image only on a 640x480 pixel computer screen, then we will gain no advantage by capturing and storing it at, say, 1024x768 pixels. To do so would waste storage space and access speed without gaining any visual advantage. On the other hand,

- ◆ At the time we capture the image, we may not know what devices will be available in the future to display it; certainly, display capability will improve during the next few years, and
- ◆ We may need to enlarge a small part of an image to examine details.

Thus, we are tempted to capture the image at the highest possible resolution, to be ready for any eventuality. However, this involves significant costs in terms of storage space and access speed, which are quantified later in the table "Storage Space and Access Time vs. Image Resolution and File Format."

Image Compression

When an image file is originally scanned, its size in bytes is:

$$\text{width(pixels)} \times \text{height(scanlines)} \times \text{color resolution(bits)} / 8 \text{ (bits/byte)}$$

For example, a 640-pixel x 480-scanline, 24-bit image has a file size of:

$$640 \times 480 \times 24 / 8 = 921,600 \text{ bytes} = 900 \text{ kilobytes}$$

(see definitions if necessary)

If even this modest file resolution were adopted for the MESL project, the 14,000 images to be contributed by the museums would occupy over 12,000MB (12GB) of disk space. Higher resolution images would require much more disk space. At 1536 x 1024 pixels (equivalent to the second highest resolution on a PhotoCD) 14,000 images would require over 64,000MB (64GB). More serious than the file server disk space requirement, however, is the amount of space that would be occupied on users' machines if they downloaded even a few dozen images to their hard disks.

To bring the storage requirements back to reasonable levels, we must compress the image files. Compression methods can be lumped into two broad categories: "lossless" and "lossy." (See Part II for more detail.) Lossless methods are so named because the restored (expanded) image is identical on a bit-by-bit basis to the original (uncompressed) image. The attainable amount of compression depends on the complexity of the image. Line drawings can be compressed greatly, but images of natural scenes can seldom be reduced by more than a factor of 2 to 4 by lossless compression methods. Lossy methods, however, can shrink image files by factors of 20 to more than 100 while preserving reasonably good image quality. The most sophisticated lossy compression methods take into account the way the human eye perceives images, so that the changes resulting from compression are not visually obvious unless the image is enlarged.

One of the most efficient available compression systems is known as JPEG. It is widely used for digital image compression, and has been adopted as the standard method for the MESL Project. Software systems that provide JPEG compression usually let you select image quality on a qualitative scale such as "Fair" to "Excellent" (e.g., in Adobe *Photoshop*) or 1 to 100 (e.g., in *WinJPEG* and *PMJPEG*). The scale only indicates a low-to-high gradation of quality. It does not indicate a compression percentage, and it does not imply a linear quality comparison. An image compressed at 80 is not "twice as good" as one compressed at 40. The quality and the amount of compression at a given setting depend on the image's complexity. At the high-quality end of the scale in Adobe *Photoshop*, JPEG is lossless or so close to lossless that the difference is insignificant for most purposes; thus, it can serve as both a lossless and lossy compression method. The available compression range is not the same in all software systems. You should experiment with your own images to determine the optimum compression ratio for your applications.

Selecting Optimum Image Resolution

Ideally, we would like to have the highest possible image resolution, so that the digital images faithfully depict the physical objects they represent and that they can be enlarged to show details clearly. However, practicality strongly limits image resolution. The table below summarizes some of the trade-offs of image resolution vs. storage requirements and image access speed.

The PhotoCD format provides a convenient framework for assessing this question. Each image on a standard PhotoCD (called a "PhotoCD Master") comes in five resolutions ranging from small "thumbnail" images suitable for multiple-image displays up to a resolution approximately equivalent to that of a 35mm slide. (The "Pro PhotoCD" format adds a higher sixth resolution suitable for medium format images, but it is probably beyond the needs of the MESL Project and will not be discussed further here.) The following table uses the five PhotoCD resolutions as categories, because they bracket the range we are interested in for the MESL Project and because PhotoCDs provide a convenient experimental medium. (See the section "Experimenting with Images" following the table.) PhotoCD images are stored on the compact disc in files having names like IMG0001.PCD. Before performing the tests in the following table, a PhotoCD file was copied to the computer hard disk. Then, images at each of the five resolutions loaded into Photoshop and then saved, first as an uncompressed Targa (TGA) file and then at the highest JPEG quality level (lowest compression) "Excellent" which is essentially lossless. Finally, using Micrografx *Picture Publisher* (which offers more JPEG options than *Photoshop*), an additional JPEG file was created at each resolution, with the quality setting chosen by trial and error to result in a compression ratio of approximately 25:1. The file sizes and display times were determined for the five image resolutions and for each compression level. Access times in this table are for comparison purposes only. You may get different results, depending on the software you are using and the configuration of your hardware. Different software systems have widely different access speeds. The terms used in this table are explained following the table.

STORAGE SPACE AND ACCESS TIME VS. IMAGE RESOLUTION AND FILE FORMAT

PhotoCD					
Designation:	Base/16	Base/4	Base	Base*4	Base*16
Resolution:	192x128	384x256	768x512	1536x1024	3072x2048
File Size (KB):					
TGA Uncompressed	72	288	1152	4608	18432
JPG "Excellent"	38	137	501	1710	6705
JPG 25:1	3	11	46	182	738

Approx Image

Access Time (seconds):

Pentium 90/48MB RAM/1GB SCSI/using Adobe Photoshop					
PCD File	Inst	Inst	6	22	133
TGA Uncompressed	Inst	Inst	2	6	21
JPG "Excellent"	Inst	Inst	4	12	40
JPG 25:1	Inst	Inst	3	6	22
Pentium 90/32B RAM/540MB IDE/using Adobe Photoshop					
PCD File	Inst	Inst	6	25	119
TGA Uncompressed	Inst	Inst	6	19	86
JPG "Excellent"	Inst	Inst	5	15	51
JPG 25:1	Inst	Inst	3	6	22
486/66/16MB RAM/2 HD totalling 1.5 GB/using PMJPEG					
PCD File	----	----	----	----	----
TGA Uncompressed	----	----	10	41	406
JPG "Excellent"	----	----	15	48	310
JPG 25:1	4	5	11	33	250
486/20/16MB RAM/1.0 GB SCSI/using WINJPEG					
PCD File	----	----	----	----	----
TGA Uncompressed	----	----	189	571	I.M.
JPG "Excellent"	----	----	251	353	I.M.
JPG 25:1	32	50	136	460	I.M.
Quadra 800/24 MB RAM/230 MB HD/using Adobe Photoshop					
PCD File	Inst	Inst	----	----	----
TGA Uncompressed	Inst	Inst	10	29	657
JPG "Excellent"	Inst	Inst	15	48	724
JPG 25:1	Inst	Inst	7	18	475
Power Mac 7100/16MB RAM/ MB/using Adobe Photoshop					
PCD File	Inst	Inst	39	134	536
TGA Uncompressed	Inst	Inst	10	27	134
JPG "Excellent"	Inst	Inst	8	34	106
JPG 25:1	Inst	Inst	4	16	50

I.M. = insufficient memory to load image

---- = test not performed

Note: The results of these tests are not intended as comparative data between machines. They are intended to show some standard hardware/software configurations that we happen to have available in our labs and to give you an idea of the kinds of performance (load/display) that you should expect for different sizes of images that you might come across during the MESL Project.

Possible Uses:	Slide sorter	Slide sorter	Full-screen viewing	Full-screen viewing	Full-screen viewing
	Text/image	Text/image	Slight zoom	2-3X zoom	4-6X zoom
			Direct video projection	Direct video projection	Slides for projection
				Slides for projection	Images for publication
					Images for publication (small)

Explanation of terms in table:

PhotoCD Designation	The terminology used by Kodak to describe the five PhotoCD resolutions. "Base" is the middle resolution. The other four resolutions are either 4 or 16 times smaller or larger than Base in terms of uncompressed image size.
Resolution	Width of image in pixels and height in scanlines.
File Size	Size of the file in kilobytes when it is stored on the disk.
TGA	Uncompressed TGA is an uncompressed file format. The sizes of the TGA files are about the same as the amount of RAM occupied by the files when they are open for viewing.
JPG "Excellent"	These represent the least amount of JPEG compression (highest quality) available using Adobe <i>Photoshop</i> .
JPG 25:1	We have found that this amount of compression results in acceptable image quality when the images are viewed on a standard VGA or SVGA computer screen.
Approx Image Access Time	The time in seconds required to open the various image formats and resolutions in different software on the indicated machines. These are typical machines, but specific performance depends very heavily on the machine configuration -- especially the amount of RAM and disk space available. Your results may differ.
Uses	Suitable applications for images of this size. (There are many others.)

Slide sorter	Display of multiple small "thumbnail" sized images resembling a photographic slide sorter or light box, to be used for quickly browsing images of interest. "Base/16" and "Base/4" images are most useful for this purpose depending on the number of images displayed simultaneously.
Text/image	Display showing a partial-screen image plus text description of the object. "Base/16" or "Base/4" images can be used. "Base/16" images are well suited to show multiple views of the same object.
Full screen	Image expanded to fill the screen. Often, a full-screen image is activated by first locating the image of interest on a "slide sorter" or "text/image" display.
Zoom amount	The amount the image can be enlarged beyond full-screen size before individual pixels or compression artifacts become obvious. This can vary considerably depending on image complexity and amount of compression.
Direct video projection	Image quality suitable for large-screen classroom projection using a digital video projector. Because of the resolution limitations of video projectors, image quality usually need not be better than that suitable or viewing on a computer monitor.
Slides for projection	Digital image quality adequate for making 35mm slides suitable projection. (Slides to be used only for projection usually have lower resolution requirements than for publication.)
Images for publication	Base*4 images may be publishable up to about 4"x6". Base*16 images may be publishable up to about 8"x10". These sizes are highly approximate and depend on the photographic quality of the image and the half-tone spacing of the publication.

Experimenting with Images

Since PhotoCDs are readily available, inexpensive, and viewable using a variety of common software packages, they provide a handy way to experiment with a wide range of resolutions of the same image. You can either obtain commercially available PhotoCDs or, better, you can have PhotoCDs made of your own objects. Existing slides or negatives can be scanned onto PhotoCDs, or you can have the images scanned onto the PhotoCD at the time the film is originally processed. The latter is less expensive per image, but you do not have an opportunity to select the best images before they are scanned. Scanning costs about \$0.60 to \$0.70 per image at the time the film is processed

and from \$0.70 to \$3.00 per image from existing slides or negatives. The higher prices are charged by professional labs who clean the film and monitor exposure and color balance. There may be an additional charge up to \$10 for the blank disk, depending on how many images are scanned at one time. Of course, you should use only high quality original images so you will be able to assess fairly the scanning and display processes. PhotoCD files (each of which contains five resolutions) are compressed using a proprietary Kodak lossless compression method. They average from 3MB to 6MB depending on image content. Each PhotoCD disk holds up to 100 images. Several sets of images can be scanned onto a given disk at different times. PhotoCD files have names like IMG0001.PCD and each disc has a unique serial number.

The best way to experiment with PhotoCD images is to copy a few of them from the CD-ROM drive to your computer hard disk. (For your experiments, it is better to load the images into the viewing software from the computer hard disk than from the CD-ROM drive so that the speed of the CD-ROM drive will not be a factor when you compare loading times on different machines.) You can work directly with PhotoCD files, but it is better to convert them to a different format such as JPEG. PhotoCD files usually load much more slowly than other formats, as you can see from the above table, and the PCD format is "read-only." You cannot save modified images in that format, but you can save them in other formats, such as TIFF, Targa, or JPEG. You do this by loading them into a software package that can access PhotoCD files, such as Adobe Photoshop (available for Windows and Mac) or any of a number of image file utility programs, such as WinJPEG. Your computer support people may be able to suggest suitable packages. You will need to check the RAM and disk space requirements for each software package. When you attempt to open a PhotoCD image, you will be asked to choose one of the five available resolutions. Pick each of the resolutions in turn, and save the image at the highest available JPEG quality level and two lower quality levels for use in your experiments (see below). Give each of the files a distinctive name that will allow you to keep track of its original resolution and its JPEG quality level. When you have created a set of test images in this way, you can transmit them over your computer network to other machines you are using for your experiments. Those machines will only need simple JPEG viewers, rather than high-end graphics packages like *Photoshop*, and they will not need CD-ROM drives.

NOTE: The act of saving a file in a different format or at a different compression level does not change the file currently in the computer's active memory. To see the effects of compression, for example, you must (1) save the image in the compressed format, (2) close the image file, and then (3) reopen the compressed image file. Also be aware that, no matter how much the file is compressed for storage on the hard disk, when you reopen it for viewing, it expands to its full original size.

First, see how much you can enlarge uncompressed "Base," "Base*4," or "Base*16" images before you begin to notice their pixel structure. Then, compare images that are identical

except for the amounts of JPEG compression. Compare the file sizes, image quality, and access speed. See how much you can enlarge the "Base," "Base*4," or "Base*16" images before you begin to notice compression artifacts such as halos near sharp edges or an overall tile-like pattern, especially when using the higher-compression/lower-quality settings.

NOTE: the two lowest PhotoCD resolutions ("Base/16" and "Base/4") are not intended to fill the screen. If they are enlarged very much, their pixel structure will become obvious.

Compare compressed JPEG images with original PhotoCD images or lossless JPEG images. Several image editing packages, such as Photoshop, allow you to open two or more image files simultaneously, so you can compare the image quality of the original and compressed images side-by-side on the same screen. (Your ability to do this depends on the amount of RAM and VRAM in your computer.)

Pay attention to how long it takes to load various image sizes from "Base/16" through "Base*16." Perform the experiments on various kinds of machines ranging from low-end PCs and Macs to Pentium PCs and PowerMacs, and especially on the kinds of machines common among your user community. Be forewarned, however, that you may not find low-end PCs or Macs that have enough RAM and disk space to load the highest resolution images.

Fowler Museum System

At the Fowler Museum we have been producing images of two resolutions using direct electronic capture methods, one from a video camera and the other from a digital camera. (In each case, two images are created of each view of an object. The larger one is used for full-screen displays and the smaller one for multiple-image ("slide sorter") and text/image displays. The smaller images are 200x150 pixels before cropping. In the following discussion, only the larger images will be described because the smaller ones do not add significantly to disk storage requirements.) We also have transferred a moderate number of photographs to PhotoCDs, and we expect to use this medium more extensively in the future.

Images From Video Camera. These images have original dimensions of 640x480 pixels. This size provides full-screen images that match the computer screen when the computer display is set to standard VGA resolution. They are slightly lower resolution than PhotoCD "Base" (768x512). They are captured using a professional Super-VHS video camera and are digitized using a Targa board. Their uncompressed image size is 900KB. Cropping brings the average uncompressed file size down to 600-700KB. The images are compressed by JPEG an average of 30:1 yielding compressed images of approximately 20-30KB. We use this capture method for objects such as sculptures that lack fine details.

Images From Digital Camera. These images have original dimensions of 1536x1024 pixels. This size provides a full-screen image with resolution about 50% better than that of the computer screen when the computer display is set to Super-VGA (1024x768) resolution. They are captured using a Kodak DCS420 digital camera and are equivalent in pixel size to PhotoCD "Base*4" (1536x1024). Their uncompressed file size is 4.6MB. Cropping cuts the average uncompressed file size down to 2.5-4MB. The images are compressed by JPEG an average of 50:1 yielding compressed images of approximately 50-90KB. We use this capture method for highly-detailed objects such as textiles. The extra resolution allows substantial enlargement of image details before the pixel structure of the image becomes obvious.

Conclusions

It might seem that the ideal image format for the MESL Project would be PhotoCD, because each PhotoCD file contains five resolutions. All users would have access to the "Base/16" or "Base/4" images for rapid browsing or text/image displays. The vast majority of users would have access to good- quality full-screen images at the "Base" level, and the few users who need high resolution would have the "Base*4" or "Base*16" images. Unfortunately, storage of 14,000 PhotoCD images for the MESL Project would require about 63GB of disk space. This is a nontrivial storage requirement even for university mainframe systems. Moreover, we have to keep in mind that the MESL Project is a test project and that the technology developed for it must be easily scalable to much larger image databases. While PhotoCD may be a very good format for transferring image files from the museums to the universities, on-line image files probably will have to be in a different format for reasons of both storage space and access speed.

As the figures in the above table indicate, it is impractical at the present state of the art to store or access large numbers of images having resolutions equivalent to the highest PhotoCD resolution (Base*16). They take up too much disk space and they open too slowly even on the fastest available personal computers, for the purposes of most MESL users. Of course, there are special applications that require high resolution images and where either the image opening time can be tolerated or only a part of the image can be displayed to examine details.

Our recommendation would be to provide:

- ◆ On-line images at the VGA or PhotoCD "Base" size for most users (together with a smaller version of each image for multiple-image or text/image displays)

- ◆ Limited numbers of higher resolution images for users having special requirements. These high resolution images probably should be provided in PhotoCD format, since it is among the most efficient lossless formats.

A clear distinction must be made between (1) the format of images actually made available on-line to end users and (2) the format used to transfer images from the museums to the universities. For the latter purpose, PhotoCD may be the most flexible, because the five resolutions allow the universities to extract whichever size they deem the most useful for their users. It is important to remember that PhotoCD image files can be copied to other less-expensive media such as tape for distribution between institutions. It is not necessary to send actual discs. They can also be transmitted across the Internet via ftp, but because of their size this is apt to prove tedious.

Our Answers to Some Questions

First, when I (Virginia) replied to some questions sent by one of the participants, I assumed that most people would have SVGA class monitors or equivalent Macintosh monitors for most of the work that would make use of the MESL images. I still believe that assumption applies. However, based on the responses of other MESL participants, I would now also assume that there will be a significant use of higher resolution (both color and spatial resolutions) images.

Also, recall that I made the assumption at that time was that it was preferable for the museums to provide multiple resolutions rather than having the universities modify the images. This assumption is based on several issues:

- ◆ Regardless of the method used to change either the color or spatial resolution of an image, once the copyright notice is applied to the image, the copyright notice will be changed in the same way as the image itself. The consequence of this is that the copyright notice is likely to become illegible in the process.
- ◆ Realistically we also need to consider how to incorporate the copyright notice when a university decides to enlarge selected portions of an image. How does this happen in a way that insures the intellectual property rights of the museums are preserved? (We do not have an answer to this just now.)
- ◆ The staff at the museums are better able to insure that the images accurately reflect the "actual" objects in their collections whereas, even with care, it is unlikely that staff and faculty at the universities are as familiar with the the features, colors, and other aspects of the objects being represented.

With these issues in mind, we now address the questions posed by our questioner.

The first question asked how we would attack the problem of reducing both the spatial and color resolution of an image that was too large to be reasonably displayed. As noted above, we believe that we also need to consider the issue of how to enlarge selected portions of supplied images as well, simply for completeness. Our questioner proposed two alternatives to accomplish each of the reductions -- pixel averaging and nth scanline/mth pixel. We will address each.

Regardless of the process used, it is extremely difficult, if not impossible, to separate the process of reducing the spatial resolution from the process of reducing the color resolution. This is true because, regardless of the method used to accomplish the spatial resolution change, there will be changes in the colors of pixels in the same relative positions in the image as the original, even if the same color resolution is maintained through the spatial resolution change.

I (Virginia) would not use either of the specifically proposed ways of changing the size. Strictly speaking, Adobe *PhotoShop* does not average pixels to accomplish a spatial resolution reduction. The alternatives available in *PhotoShop* are bicubic, bilinear, and nearest neighbor. Of the choices available in *PhotoShop*, I would choose bicubic. Corel's *PhotoPaint* offers anti-aliasing and stretch/truncate as the methods. Of these two choices, I would select anti-aliasing. In the greater scheme of the world and not limited to *PhotoShop* or *PhotoPaint* as my tool, I would use either cubic convolution, bilinear convolution, or nearest neighbor methods (my preference is reflected in the ordering). I have found all of these are much better at producing a reasonably sized and reasonably colored image than either a strict nth line/mth pixel or pixel averaging method. All three are good at processing "natural" images, i.e., ones which do not tend to have geometric constructs in them. Additionally, they tend to minimize digital artifacts for images with geometric objects.

In theory, the method of reducing the size is independent of the method used to store the images. In practice, you might have to use one software program to accomplish the reduction and another to store the image simply because most software vendors do not incorporate the concept of maximum flexibility in their products. For enlarging selected portions of a supplied image, I would use cubic convolution in preference to any other method

Why do I (Virginia) have these preferences?

For enlarging processes, the reverse process of the nth scanline/mth pixel reduction method our questioner mentioned is called pixel replication. In this process, pixels are duplicated, in line, and then the lengthened scanline is replicated. This usually results in a pixelated image, i.e., an image where the square shape of the pixels becomes really evident to the viewer.

If you take a look at images containing geometric constructs such as outlines of circles or squares/rectangles which are canted to an odd angle, you will find that lossy JPEG images tend to create (and sometimes perpetuate) digital artifacts (things that are present in the reconstructed image simply because of the method used to store the image). Good examples of things that generate these kinds of artifacts (and that I have fiddled with) are paintings, etchings, and sketches by Kandinsky, Klee, and Miro, for example. By contrast, the things I referred to as "natural" scenes are images more like say, Van Gogh's *Iris*, Manet's *Salmon*, *Pike*, and *Shrimp*, or even Rembrandt's *Self-Portrait*.

I have found that when I have a lossless JPEG image coming in, all three methods I identified above (nearest neighbor, bilinear convolution, and cubic convolution) did a very satisfactory job of creating good 640x480 or 1024x768 8-bit or 24-bit images from a 1280x1024 24-bit image. They also worked well with a very large source image having an original size of 6Kx3K.

The software I personally would use is real image processing software, not a PC paint package. This software originated in various places, primarily NASA and the Department of Defense, and generally runs in OS/2 (because that is the environment I have ported it to) or on various flavors of mainframe and UNIX boxes. If constrained to a PC-based package, I would probably choose among Adobe's *PhotoShop* or Corel's *PhotoPaint* among the commercial products. Of the non-commercial products, I would probably choose among *ImageMagick* which was originated at DuPont, *pbmplus* from Jeff Pozkaner, and *Image Alchemy* which was developed by a private individual. You might also want to investigate *Khoros* (for Unix systems) which is a shareware package from the University of New Mexico.

The bad part about any method is that, if it is presented with a lossy image, particularly one that is especially lossy, it can give just as ugly a result as the two methods mentioned in the questions.

Now comes the issue of color resolution. Nearly all display systems include a mechanism for dithering an image when the color resolution is too great to represent accurately. My personal inclination is to allow the systems themselves to deal with the color resolution issue rather than to try to account for all the variability in the images from the museums. As a minimum, we might expect the museums will provide options for 8-bit and/or 24-bit images of their objects. I (Virginia) expect that the universities will wind up using the 8-bit images much more frequently than they will use the 24-bit images just because it will be easier to use the 8-bit images on the video projection systems usually found in general assignment classrooms.

As far as color palette truncation is concerned, I (Virginia) believe that the better choice is to dither. Color palette truncation generally results in extremely unsatisfactory results

unless the person creating the truncated palette has more than half-vast experience with this process. Color palette truncation is usually specific to each image. This would require a huge increase in the amount of time needed at the museums to create the images.

So, from my (Virginia's) perspective, having done image processing work on just about every non-classified system available at NASA, the DoD, and the universities where I have worked over the past twenty years, I really believe that the best thing is to decide what resolution is required and have the people who are most familiar with the materials being imaged actually do the work. This has the added advantage that the copyright notations can be put on after the resolution work has been accomplished so that the copyright notation does not get munged (a technical term meaning mashed until no good). I would be willing to wager (maybe as much as a nickel or so) that nearly every museum has a very high priority on the copyright issue.

Now, having said that, we also see that an absolutely legitimate educational use of the images would be to enlarge certain parts of an image to illustrate particular points for students. It is also probably unrealistic to expect that the museums would actually have the time and staff to supply specific enlargements of this type. So, someplace we need to consider whether we need to include the copyright notice on the enlarged image, and, if so, how to do it, including the form of the copyright notice as well as the actual position on the image.

Having outlined some issues here, obviously different institutions will have different views of the points we have raised.