Web Imaging Secrets

I have sure been having lots of fun lately with my new *scanner method* digital "photos". By using a scanner as a digital camera, you easily end up with *hundreds* of times the resolution. Which gets especially important for web product photos, images of electronic test equipment or wherever clarity and lettering detail is essential.

We saw more on the scanner method in IMAGIMAG.PDF and MUSE140.PDF. Some stunning examples of the results are found in *www.tinaja.com/barg01.html* .

But while all this was happening, a neighbor drops in with a "Duh - How do I get a picture onto my eBay sale?" question. So, what I thought I'd do here is go over a few fundamentals of creating web images and then take you step-by-step through some scanner method secrets.

Image Data Formats

The arrangement of bits and bytes of visual image info into a file is known as your *image data format*. A chosen data format trades off file *length*, *clarity*, *accuracy*, *speed*, *colors*, *artifacts*, *resolution*, *appearance device independence*, *size scability*, *transparency*, and *editibility*. Before you begin to think about putting your image on the web, you first have to understand the major image formats...

Bitmaps–A bitmap is just that: a sequentially ordered map of bits. Some number of bits represent *one* image pixel Perhaps one bit for "litho" black and white drawings. Or 24 bits for three colors of 256 possible levels each.

Bitmaps work best at one precise resolution. While you can double or halve that size, casual size adjustments often will trash your result. Especially rotations. Bitmap file sizes can end up outrageously enormous. Much more so should you mistakenly overspecify the needed resolution or the number of required color bits.

Bitmaps usually will be your preferred choice for initial "photo" image gathering and editing because they are both accurate and artifact-free. Especially when combined with CD-R data storage. But their excessive file sizes makes them generally unsuitable for direct web use.

GIF– The popularity of this format is waning, but it still is a very compact way to accurately show sharp lettering in a limited number of colors. This format is easy to animate. GIF files are primarily used today for banners or to make websites more obnoxiously annoying. Shareware programs such as the *GIF Construction Set* can be a great help here. There are also great heaping piles of GIF web resources. You can reach these through *www.tinaja.com/webwb01.html*

JPEG– JPEG is a method to compress image bitmaps. In the process, your file sizes do get a lot smaller, but your image usually degrades and noticable artifacts may appear. JPEG is based on a hairy piece of math called the *Discrete Cosine Transform*. Pictures are broken down into 8x8 pixel blocks. Blocks "close" to each other visually are approximated with common blocks. The image then gets reassembled into a much shorter file. JPEG is probably best for camera-style images or for selling stuff on eBay.

Acrobat PDF– That general purpose PostScript computer language can create visuals using simple descriptions. Such as moving to locations and then drawing lines or curves.

Given only the needed control or end points as data.

Such visual graphics are extremely compact. They easily adjust to any device resolution. They may be magnified or rotated at will. Or repeatedly reused. For those cases where only a photographic style image will do, bitmaps or JPEG are readily importable. But careless use of these imports may severely bloat your code.

The advantages of Acrobat include portability and device independence combined with a precise control of exactly what your end user will see. All done in extremely compact file sizes. PostScript can be hand generated, or you might "invisibly" use PostScript inside of *Illustrator* or *Pagemaker*. Adobe Acrobat is a compact and elegant "pseudocompiled" runtime form of PostScript. A readily available free viewer is needed to use Acrobat files.

There are many other possible image data formats (fax, CMYK, MPEG, fractal, runlength, wavelet, TIFF, and scads more), but these we just listed are the mainstream for web authoring and distribution. Programs like *Image Alchemy* or *debabelizer* let you convert between the more exotic and more popular data formats. I'll normally use *ImageViewer32* to get between major formats.

My recommended data format choices are as follows: When and where possible, always use Acrobat PDF files for web distribution. Making as much of as many of them ultra compact procedural code instead of embedded images.

For instance, an electronic data sheet could take 15 Megs as a scanned bitmap. Or 150K in JPEG, Or less than 9K in a vastly better looking, device independent, and easily scaled PostScript language PDF description. More details are found in *www.tinaja.com/acrob01.html*, in *www.tinaja.com/weblib01.html*, and in *www.tinaja.com/post01.html*.

Use GIF files for small ad banners of limited colors where lettering legibility is super important. And possibly for bars

and buttons and similar animated bells and whistles.

When only a photo will do, raw bitmaps are by far your best choice for initial editing. For general web "photo" use, reducing bitmaps to JPEG format is often a good route.

But only as a last out-the-door editing step.

Because web viewers are impatient, it always pays to use click-expandable images or "click for photo" buttons.

For *eBay*, JPEG images are probably your best choice, since there are restrictions to when and where and how you can use the other formats.

Getting your Image onto the Web

Step one of putting an image onto the web is finding a place to put it. Step two is telling yourself or others how to *link* to that image. Many newbies are surprised to find out that eBay does *not* store images for them! All eBay does is stash ultra brief *links* to your requested images.

Which *must* be placed elsewhere on the web.

You normally do not stash an image on your home PC to make it web available. The many reasons include having to keep your computer and phone line open 24 hours a day, an uninterruptable power supply, and those $800 a month or so charges for direct web backbone access.

Instead, you go to a local ISP Internet Service Provider or other service who *hosts* your website for you. Your website url may get added to theirs (*www.zekes.com/~mygreatsite*). Or, with *virtual hosting*, you can register your *url* with *Internic* (*www.mygreatsite*) This latter url name is uniquely yours, shorter, and fully portable. But costs $50 per year.

Thus, to use eBay, you must have your own website and home page. Or else "borrow" space elsewhere.

At any rate, your website sits in your ISP's computer, not yours. Your site normally consists of a home page, support pages, and subdirectories that hold useful files for you. It is a good idea to create an *Images* folder specifically to hold all of your web pictures for you.

Your *NetScape* or whatever browser is great for viewing your own or other websites. But it cannot *change* your site or add to it. Instead, special password protected software is needed to move files on to or off of your website.

Such software is usually given you by your ISP. An older FTP or *File Transfer Protocol* format is normally still used. I use John Junod's WS_FTP32 shareware. This gives you two side-by-side catalogs that let you move files from or to your website from your home computer.

Thus, the usual way of putting an image onto the web is to move it from your home computer to your ISP host computer with FTP shareware. Once there, you can *link* the image onto other pages of your own by using *NetScape* or other page creation software. Or you can tell others about your image by giving them the address.

Which might be *www.zekes.com/~mygreatsite/images/sup erpix.jpg* for a normal hosting or *www.mygreatsite/image s/superpix.jpg* for a virtually hosted site.

Some eBay Specifics

Getting your photo on eBay is simple enough. First, you get a photo into a suitable image data format on a home computer. Second, you use a FTP file transfer program to move a copy of your photo to a net accessible site. Such as in an *images* folder on your or some other webisite. Then you tell eBay how to reach it. You do this as a normal part of entering your for-sale auction data. Your first image must be JPEG or GIF. JPEG is best here.

But you can easily get a lot fancier. To do this, you have to learn a little about a language called HTML. In general, an HTML command is bracketed by <carats>. A forward slash stops a previous HTML action, while its absence starts one. Many begineer's books on HTML can be found at my *www.tinaja.com/amlink01.html*

You are allowed to embed certain HTML commands in your eBay product description. Here is just enough HTML to get you in trouble on eBay…

To add a second or higher photo:

<IMG SRC="your_image_url_goes_here"></A>

To add attractive paragraph breaks:

Start each paragraph with <P>. End with </P>.

To link a web page, animation, or sound:

<A HREF="exact_url_here"> put description here </A>

To precisely arrange multiple photos:

Learn about HTML tables and how to use them.

Be sure to carefully preview your eBay entries any time you try to add anything fancy in the way of HTML. Do make sure all of your links and commands are doing what you have asked of them.

Using my Scanner Method

Let's go through a detailed description of how to get from an object to a web image. We will use the Sony still video player of *www.tinaja.com/images/bargs/vidfls01.jpg* as an example. I challenge you to get results even remotely this good by any other method.

For large or awkward objects, digital cameras may be the way to go. But I prefer the scanner method because of the outstanding resolution and image quality it can offer.

The advantages of the scanner method lie in its clarity and sharpness. Exposure and focus and lighting are nearly always correct. You are obviously limited to "straight on" views of managable sized objects. The scanner's amazing but not infinite depth of field restricts certain objects. Or needs multiple exposures or software adjustment.

My favorite tools are a *HP Scanjet* 6100C with support software. Rather than use the obvious *Photoshop*, *Freehand*, or *Illustrator*, I prefer to work with cheaper and lower level programs. I use Arcata Pet's *Imageviewer32* shareware to get between bitmap and JPEG formats. And to alter brightness, contrast, gamma, sharpness, color balance, cropping, and orientation. Plus plain old *Paint* for super accurate pixel level editing. And, of course, *Acrobat Distiller* to get from PostScript to PDF.

Use of large bitmapped files can be eased with CD-R storage. However, I've found the *Adaptec* software blows up and destroys disks repeatedly when you try to use large bitmaps in Paint with it. These problems seem to go away if you…

(1) Save to CD-R only *after* closing Paint.
(2) Avoid repeated use of filename overwrites.
(3) Store one or two jobs per disk at most.
(4) Always eject your CD-R disk immediatley.

The latter forces closure any active files, instead of them hanging around where they can be trashed.

Back to the program. The HP scanner is run in its "Sharp Millions of Colors" mode with the "Emphasis" tool selected as "Lighten Midtones" and applied. The cover gets removed only during actual use. The scanner gets run in its "Better Illumination" slow warmup mode. Its glass gets carefully cleaned *immediately* before each shot.

The Sony unit was a little too large to fit the scanner, but not big enough that multiple exposures were needed. I just made sure one edge was scannable. The item being scanned also must get cleaned "good enough" for legible lettering. Note you can easily retouch defects or labels or supports or whatever. Speaking of which, *minimum height* supports made from erasers or whatever are normally used to get your "main" item surface parallel to the scanner glass.

But no props were not needed for this subject.

It is extremely important to use a machinist's square to align the object with the scanner. Always aim for a one pixel accuracy of all edges!

Normally, you'll want to zoom "somewhat". This adds clarity and gives a near 1:1 size with higher resolution web displays. In this case, the object was too big to zoom. You also should *not* zoom multiple exposures.

The scanning direction can affect whether shadows end up in the lettering or not. Your choice of object centering can minimize "Barney Oldfield" effects where knobs lean asymmetrically like the wheels on early race car photos. I always prefer to scan sideways or upside down. This lets me concentrate on editing detail, rather than subject.

Your captured image is saved both to an internal drive and to CD-R. It is then brought up into Paint. In this case, we need a larger bitmap to make room for the missing edges. So Paint's *Image Attributes* were expanded by a few hundred pixels and the image recentered.

A bright purple (or otherwise highly legible) line is then drawn near an edge to verify that you are within one or two pixels of "perfect" alignment. If not, you reshoot. I prefer to force true pixel edge justification because edges are the first thing your eye sees.

Some Editing Details

A slightly mottled but otherwise undistinguished pastel background is created and then gets expanded around but not touching the subject. The mottling helps bunches in getting rid of the more annoying JPEG edge artifacts.

The middle of a subject edge is then selected and the background brought close enough to make a "credible" edge. Jarring or disconcerting pixels can be removed by replacing them with nearby ones. The object is to create a largely nondescript but precise looking edge.

You then "slide" this edge up or down (or right and left) to replace your entire original edge with a clean and pixel aligned one. Grab a piece of edge, rotate it ninety degrees and work across the top. Come back and fix the corner.

Now, copy your corner, rotate it ninety degrees and go do the next corner. Work your away around till you have the outside object edges pixel accurate. If you get a pixel or two out of step, you can jink the entire remaining image up or down to get perfect alignment.

You then edit your way in from an edge, removing any blemishes and improving secondary edges with a similar "grab-adjust-slide" process. Some pixel-by-pixel retouching might be needed around lettering.

Additional tricks may be needed on other subjects. I like to crowd shadows or eliminate them entirely. Do this by expanding lighter areas so your last "brown" line overwrites most of the "black" shadow lines. Use of symmetry can very much brighten and improve round objects.

Sometimes you actually want to do certain details from scratch, making them pixel accurate. Such as the tick marks on a scope screen, threads, or precise probe details. But try not to be too obvious. Aim for "continuity" between your original scan and any fabricated new stuff.

Remember that the eye is better at seeing detail than color, so shape counts the most.

Retouching lettering is best avoided, because it is likely to end up obvious. If you must, recreate the lettering and replace it. I do this by using raw PostScript and Acrobat for larger letters or fancier fonts. Or by hand creating 5x7 3x5 or even 2x3 pixel maps for the really small stuff.

Details can get addded by pasting in secondary scans. Especially any wires or connectors. Feet and grommets and such are sometimes best done simply by drawing them in Paint. Fuzzy or dark areas can be moved into ImageViewer, sharpened, lightened, and replaced. What the photo folks used to call dodging and burning. But a little of this goes a long way. To avoid any "sugar" on sharpening, a good rule is to *take half and leave half.*

Detail work is best done on the side. Grab what is to be fixed, drag it into the background, fix it, and put it back.

This avoids embarassing surprises.

Once you have your edited bitmap the way you want it, you resave it. Then drag it into *ImageViewer32*, crop it to a nice background size, and rotate it right side up. Usually I'll brighten *slightly* by a notch or two at this point as well. Gamma and contrast retouches can also be explored.

You then save as a web-ready JPEG image. Finally, you FTP your image to your web site.

The Ultimate Test

The ultimate test of a good picture editing job is simply this: Nearly all of your effort should seem to vanish in its entirety! The best of editing draws no attention whatever to itself. Leaving you with a "pure" image.

One warning: My scanner method can end up incredibly addictive. Avoid spending hundreds of detailing hours to sell one $5 item. Know when to quit. Unless you're having fun or using the time for skill building.

For More Help

Repeating some of our previously mentioned resources, more info on the scanner method can be located in my IMAGIMAG.PDF and MUSE140.PDF. Dozens of real world examples are in www.tinaja.com/barg01.html and related pages. Our own eBay auctions are at http://cgi.ebay.com/eBayISAPI.dll?ViewListedItemsLw-cgi/elinkBUttons&userid=abeja

More on web images at www.tinaja.com/acrob01.html, at www.tinaja.com/weblib01.html. More on PostScript usage at www.tinaja.com/post01.html Books on webmastering, HTML and related topics at www.tinaja.com/amlink01.html

I also do have training seminars available on the scanner method and similar topics. Often held at that Black Range Lodge craftily hidden in the secret part of New Mexico that you can't get to. See www.tinaja.com/info01.html and my www.tinaja.com/consul01.html for details.

Finally, through Bee's Abeja Designs, we are now available to offer cash-and carry scanner method image creation at a flat $99 per appropriate subject.

Let's hear from you. ❖

PLEASE CLICK HERE TO...

▶ Get a Synergetics catalog

▶ Start your tech venture

▶ Sponsor a display banner

▶ Find research solutions

▶ Send Don Lancaster email

▶ Pick up surplus bargains

▶ Find out what a tinaja is

▶ View recommended books