

Transcoding Characteristics of Web Images*

Surendar Chandra, Ashish Gehani, Carla Schlatter Ellis and Amin Vahdat
Department of Computer Science, Duke University, Durham, NC 27708-0129
{surendar, geha, carla, vahdat}@cs.duke.edu

ABSTRACT

Transcoding is a technique employed by network proxies to dynamically customize multimedia objects for prevailing network conditions and individual client characteristics. Transcoding can be performed along a number of different axes and the specific transcoding technique used depends on the type of multimedia object. Our goal in this paper is to understand the nature of typical Internet images and their transcoding characteristics. We focus our attention on transcodings intended to customize an image for file size savings. Our results allow the developers of a transcoding proxy server to choose the appropriate transcoding techniques for the important classes of Internet images.

We analyze the characteristics of images available on the Web through a representative trace. We show that most GIF images accessed on the Internet are small; about 80% of the GIF images are smaller than 6 KBs. JPEG images are larger than GIF images; about 40% of the JPEG images are larger than 6 KBs. We also establish the characteristics of popular image transcoding operations. We show that for JPEG images, the JPEG compression metric and a transcoding that reduces the spatial geometry are productive transcoding operations (saves at least 50% of the file size for 50% of the images). Our systematic study of image characteristics leads to some surprising results. For example, a naive spatial geometry reduction of GIF images by a factor of 2 along each axis actually causes to an increase in the file size compared to the original image for 40% of the images. Thus it is important to understand the characteristics of individual images before choosing the proper transcoding operation.

1. INTRODUCTION

The wide availability of inexpensive hardware such as personal computers, digital cameras, scanners and other easy-to-use technologies is making it easier for the average user to dabble in multimedia. The phenomenal growth of Internet technologies such as the Web and electronic mail allows users to disseminate and share these multimedia objects. The trend for Web content providers is toward increasingly multimedia rich experiences for the end user. While the ability to share multimedia objects makes the Internet more valuable to consumers, the capabilities of the clients and the underlying network are not always able to keep up with this shifting demand.

Users access multimedia objects from a wide variety of devices with different resource constraints. Users are not only accessing multimedia objects from traditional desktops, but they are also using mobile devices such as palmtops and laptops as well as newer devices such as webtops and navigation systems that are resource constrained in terms of their processing, storage and display capabilities. Similarly, the network capabilities used in accessing these multimedia objects vary widely from wired networks such as high speed LANs, ISDN, DSL, cable and telephone modems to wireless technologies such as cellular, CDPD, Ricochet and GSM networks. These networks exhibit wide variability in speed, reliability and cost.

In such an environment of varying network, storage and display capabilities, there is a need to customize the object for the prevailing operating conditions. Consumers using expensive networks may wish to download multimedia objects for the lowest possible cost. On the other hand, consumers using high-speed networks and high-quality displays may desire to view the images at the highest quality. Finally, consumers using slow networks want to view images as quickly as possible.

Transcoding is a transformation that converts a multimedia object from one form to another, frequently trading object fidelity for size. By their very nature, multimedia objects are amenable to such soft access where users are willing to compromise object fidelity for faster/cheaper access. Transcoding has been used in a number of systems¹⁻¹⁶ to serve the same underlying multimedia object at different quality levels to different users based on the client operating constraints. Transcoding has also been used by Web servers¹⁷ for bandwidth consumption control and to provide differentiated Quality of Service for different clients. Some of the transcoding choices for images include color reduction, spatial size reduction (e.g. thumbnails), increasing

*This work was supported in part by a graduate fellowship from North Carolina Networking Initiative (NCNI) and equipment grants from Intel Corporation and the National Science Foundation (CDA-95-12356). The NLANR squid proxy trace logs were made possible by the National Science Foundation (grants NCR-9616602 and NCR-9521745), and the National Laboratory for Applied Network Research.

lossy compression, transliteration to audio and text. Though transcoding has been used to target a multimedia object for a particular device, in this work we are concerned with intentional transcoding to save file size.

Our goal in this work is to understand the nature of typical images in the current Internet workload and their transcoding characteristics. We focus our attention on transcodings that customize an image for file size savings. Such knowledge enables service providers to choose potential transcoding techniques that offer benefits for a wide variety of images. Such knowledge also allows the service to avoid choosing transcoding techniques that might appear promising but are not effective for their target workload. For our experiments, we analyze GIF^{18,19} and JPEG²⁰ images collected from the NCAR NLANR proxy access logs.

We show that most of the GIF images are small; about 80% of the GIF images are smaller than 6 KBs. About 45% of these GIF images appear to be bullets, icons, lines or banners. On average JPEG images are larger than GIF images, 40% of the JPEG images are larger than 6 KBs. About 30% of the small JPEG images appear to be bullets, icons, lines or banners. About 35% of JPEG images (78.3% of JPEG data) and 10% of GIF images (45.1% of GIF data) are LargeTrueImages as we define them. These image characteristics have implications for transcoding techniques.

Our evaluations of image transcodings show that there is significant opportunity for sophisticated transcoding of JPEG images. We show that for JPEG images, the JPEG compression metric and a transcoding that reduces the spatial geometry are productive transcoding operations. Since the compression metric loses visually imperceptible information first, it is a good transcoding that reduces image file size, sacrificing as little visual information as possible. Unless proper transcoding parameters are chosen, traditional transcoding techniques such as thumbnailing of GIF images have the potential of actually increasing the output image file size. For a transcoding that reduces the image by a factor of 2 and 4 along each axis, 40% and 2% of the GIF images transcode to a size that is larger than the original image file size, respectively. Hence it is important to understand the transcoding characteristics of images in order to choose the appropriate transcoding operation.

The remainder of this paper is organized as follows: Section 2 describes our experimental setup. Section 3 describes the characteristics of the images and Section 4 describes the results of performing various transcoding operations on the images. Section 5 describes related work. We conclude in Section 6.

2. EXPERIMENT OBJECTIVES AND DESIGN

2.1. Objectives

We designed our experiments to answer the following questions:

- What are the characteristics of the images accessed on the Internet today?
- How do various image transcoding techniques perform for these images? We are mainly concerned with transcodings that yield file size savings (as opposed to transcoding to fit an image to the target device characteristics). Hence, we define a transcoding technique that saves at least 50% of the file size for 50% of the images as a *productive transcoding*.

2.2. Image Collection

A workload of typical images accessed on the Web is crucial for drawing realistic conclusions about the effectiveness of a transcoding operation. However, standard access trace collections from network proxy and client Web traces^{21,22} are either too old or hash the URL's accessed so as to protect the privacy of the participants. For our study, we need the actual images so that we can measure the applicability of transcoding to a real workload and not to some synthetic images that were generated to be of the same size as the images available in an access trace.

Hence, for our experiments, we use the Squid proxy logs from NLANR.²³ NLANR proxy caches make their traces (with actual, unscrambled URL's) publicly available for the most recent seven day period. For our study, we utilized access traces that were collected on Sep 27, 1999 from the NCAR proxy at Boulder, Colorado. Of the images in the trace log, 74.81% of the unique images were GIF images and 24.41% were JPEG images. Other image formats, including PNG, make up the rest of the (0.78%) requests. GIF and JPEG images comprised 48.61% and 50.88% of the image bytes transferred, respectively. Interestingly, while most of the images are GIF's, bytes transferred are evenly distributed between GIF's and JPEG's.

For our experiments, we downloaded 97,990 GIF images and 37,616 JPEG images, totaling 402 MB and 440 MB respectively, identified from this NCAR cache access log collected on Sep 27, 1999. We tracked the various top-level Internet domains from which our images came. Not counting repeated accesses to the same image, we found that the vast majority of the images in our collection (80% of GIFs and 77% of JPEGs) are from the .com domain.

For our experiments, we used utilities from NetPBM²⁴ and libjpeg²⁵ to perform the various transcoding operations on GIF and JPEG images.

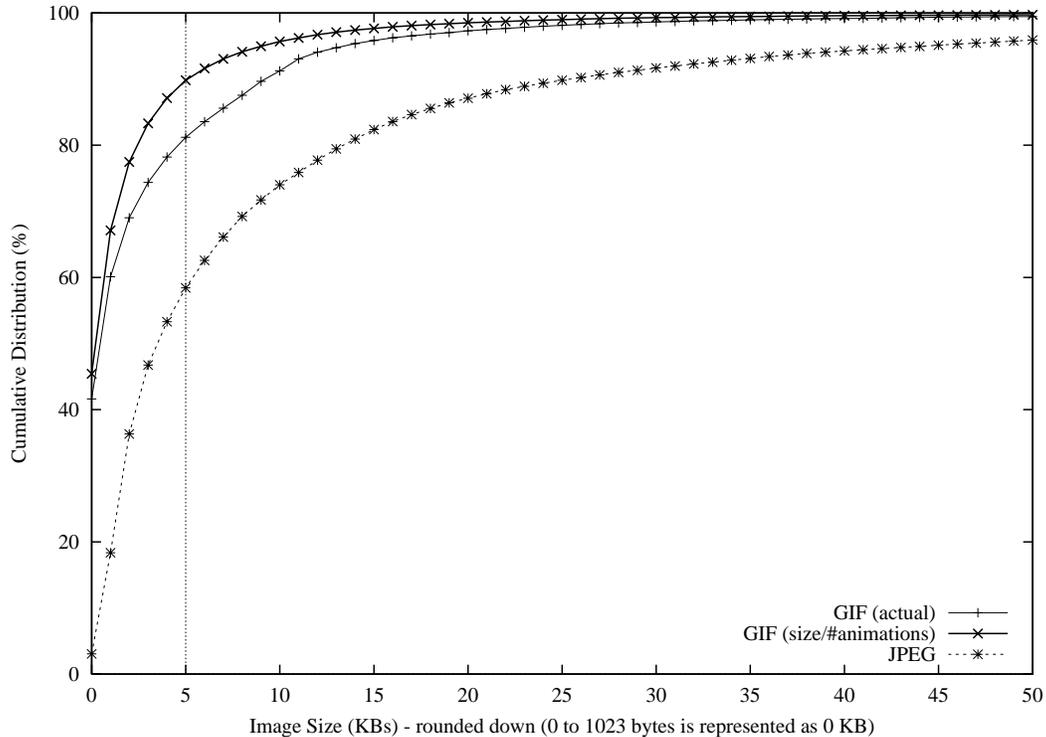


Figure 1. Image file size distribution

3. STATIC IMAGE CHARACTERISTICS

In order to understand the characteristics of images accessed on the Internet today, we first analyze the static characteristics of the images. Static image characteristics such as image geometry, file size, etc. give a sense of the possible axes along which transcoding can take place. In the next section, we analyze the transcoding characteristics for the images. We place emphasis on the characteristics of transcoding operations that apply to our image collection based on these static image characteristics.

3.1. Image File Size

We first analyze the file sizes of the images in our workload. We plot the image file size distribution as a cumulative distribution in Figure 1. From Figure 1, we note that most (about 80%) of the GIF images are small (< 6 Kilobytes). By contrast, about 60% of the JPEG images are smaller than 6 Kilobytes. Since GIF images may be animations^{18,19} where a number of individual images are packed into a single GIF image file, we analyzed the number of animations in GIF images. We found that 14% of the GIF images contain more than one animation frame. 1% of the GIF images had more than 20 animation sequences per image. An obvious way to deal with large GIF files composed of numerous animations is to reduce or eliminate all but one of the individual animation frames. Next we plotted the average size of an animated frame in a GIF image [measured as the *image file size / number of animations*] as a cumulative distribution in Figure 1. From Figure 1 we note that about 90% of the GIF images have frame size less than 6 KBs.

Traditional human factors research²⁶ has shown that the response time for accessing a resource should be in the 1 to 10 second range for information to be useful. If the response time is longer than this range, the users tend to lose interest and go on to other things. For our work, we choose a response time of 5 seconds as our preferred latency. We estimate the file size limit for images to be served within 5 seconds in the Internet. We assume a minimum 9600 baud network connection. We compute the image size that can be served within a latency of 5 seconds on a 9600 baud modem to be 5.9 KB. For many Internet usage scenarios, it is less interesting to transcode images less than 6 KB in size. We thus conclude that transcoding is not necessary for over 80% of the GIF images and 60% of the JPEG images.

3.2. Number of colors

We next analyze the number of unique colors in an image. GIF compression is optimized for line drawings with a small number of colors. Hence, reducing the number of colors is a potential transcoding for GIF images. For the GIF images, we plot the

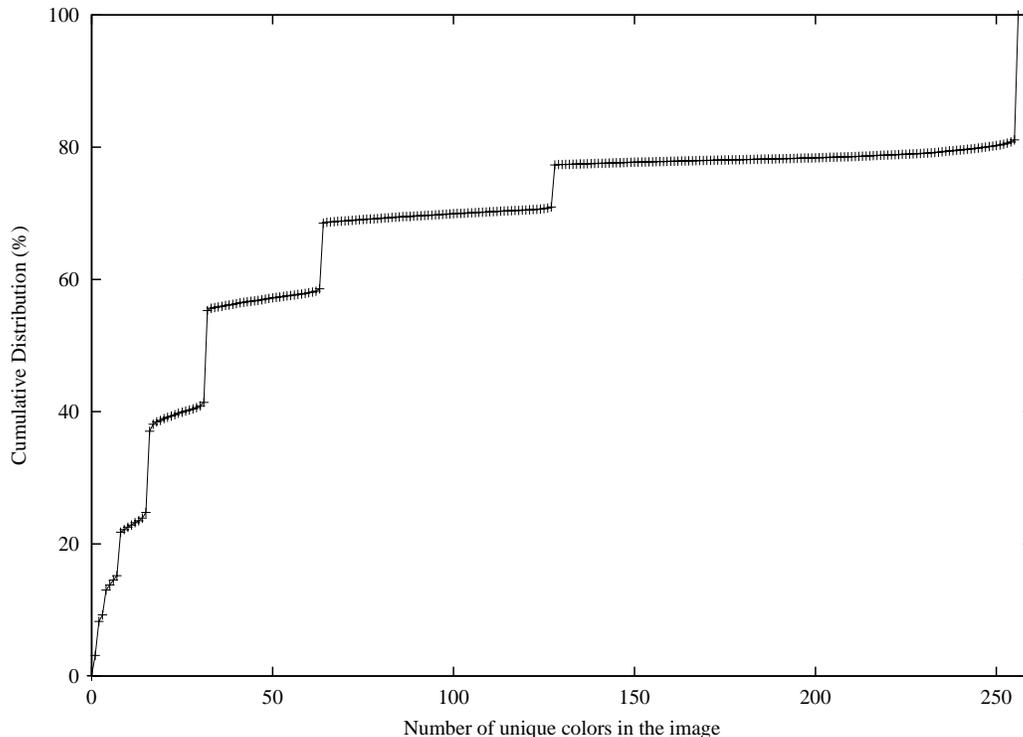


Figure 2. Distribution of number of unique colors in GIF images

number of unique colors in each image as a cumulative distribution in Figure 2. GIF specifies the number of bits required per pixel in an image in its screen descriptors. This restricts the number of unique colormap choices to a power of 2, which is evidenced in the clustering of the number of unique colors (Figure 2) Also note that at least 20% of the GIF images have 256 unique colors suggesting that these images may be photographs (which are better suited to JPEG compression).

On the other hand, the number of unique colors in an image compressed using a lossy compression technique such as JPEG not only depends on the image, but also on the precision of the decoder. Integer round-off errors introduced by the decoder add imprecision in the decoded color values. The exact number of colors does not provide much information about the original image and hence the number of unique colors for JPEG images is not explored here. JPEG is optimized for photographs and defines two color modes, TrueColor and GrayScale. For the images in our collection, 5% of the JPEG images were GrayScale.

3.3. JPEG Compression Metric

Lossy compression techniques such as JPEG define a way to specify the amount of “lossiness” using metrics such as the compression metric (also referred to as the JPEG Quality Factor by Independent JPEG Group²⁵). The Quality Factor value that was used in encoding a JPEG image is not stored along with the image. In our earlier work,²⁷ we developed an algorithm to accurately measure the initial JPEG Quality Factor of an image. Using our algorithm, we calculate the initial Quality Factor for the JPEG images in our image collection. The results are plotted as a cumulative distribution in Figure 3. From Figure 3, we note that about 50% of the JPEG images have Quality Factor values greater than 75.

3.4. Spatial geometry size

Thumbnailing images is a popular transcoding operation. For thumbnailing to be useful, the original images themselves should be above a threshold minimum dimension. Hence, we analyze the spatial geometry characteristic for our collection of Internet images. For the images in our collection, the spatial size of the images (width vs height) for GIF and JPEG images are plotted in Figure 4 and Figure 5 respectively.

From Figure 4, we note that a significant proportion of GIF images are small (less than 150x150 pixels). Also, many GIF images are wider than taller (aspect ratio of width to height greater than 10:1). From Figure 5, we note that JPEG images have a rectangular aspect ratio (1:1, 2:3 and 3:2 appear to be popular) as compared to GIF images.

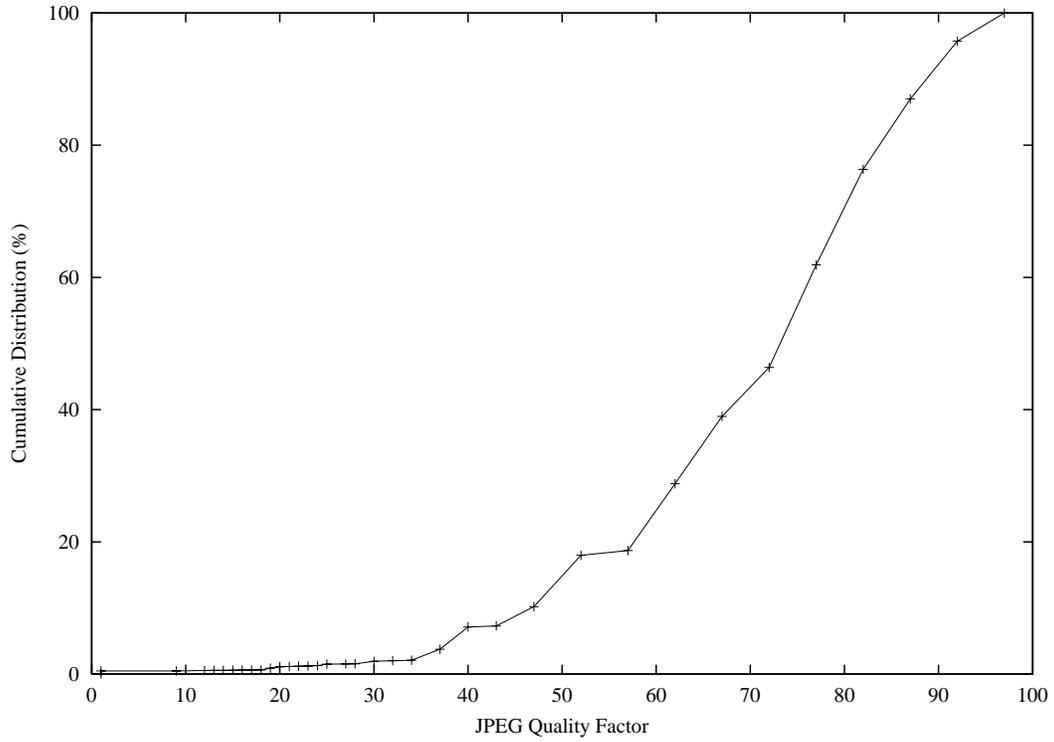


Figure 3. Initial Quality Factor of the JPEG images

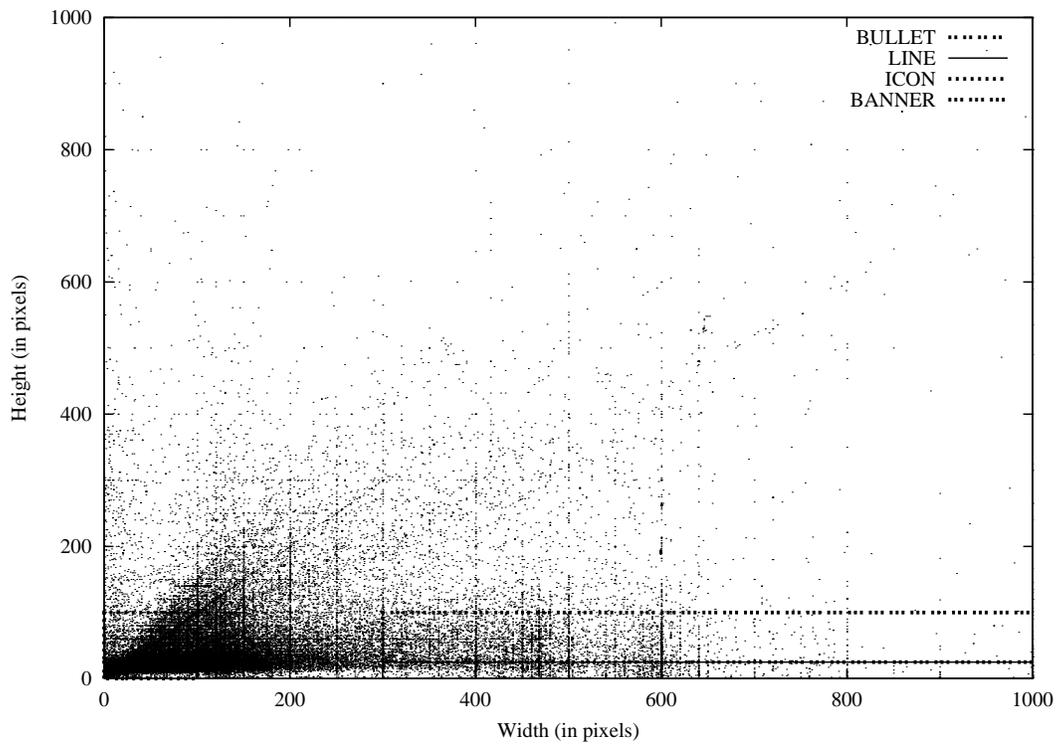


Figure 4. Image spatial size distribution - GIF images

We classify images based on the following heuristics:

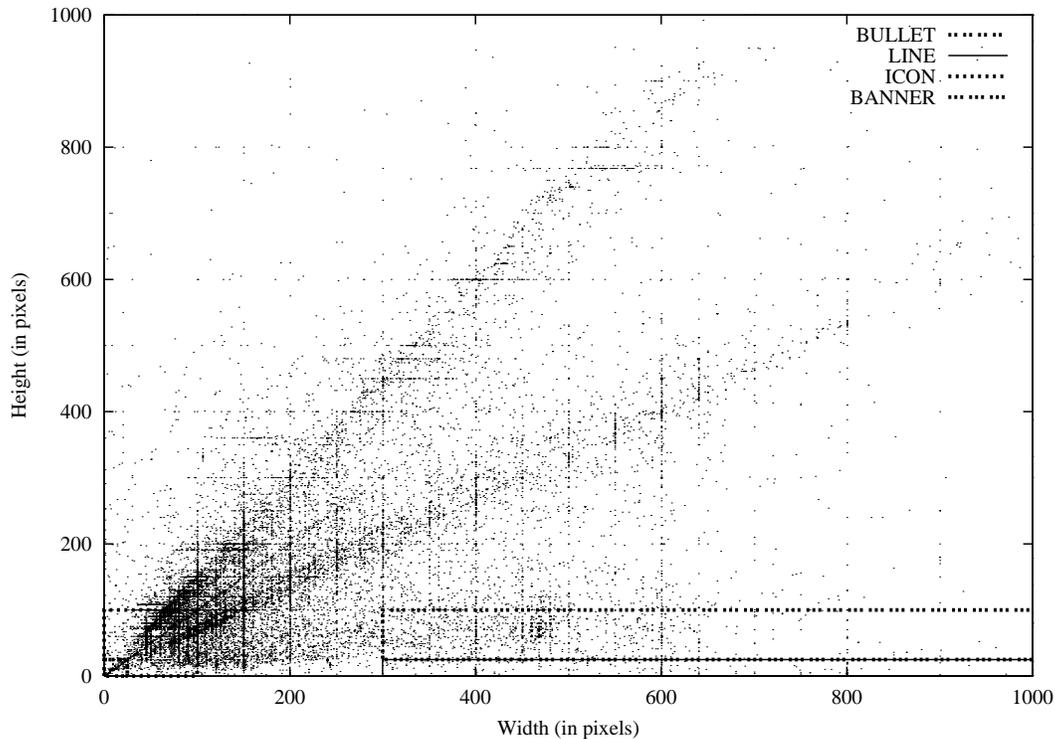


Figure 5. Image spatial size distribution - JPEG images

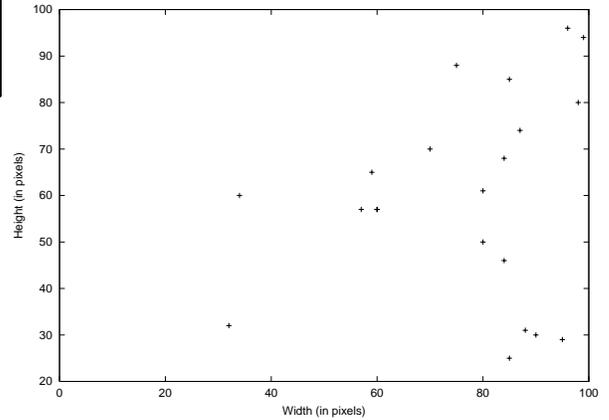
- **Bullets** - images that are smaller than 25x25 pixels are categorized as bullets. Even though the images might themselves contain high quality images and arrows, such small images convey little useful information. In situations in which transcoding is needed, these can usually be replaced by the HTML LI tag.
- **Lines** - images whose width is greater than 300 pixels and height less than 25 pixels are deemed to be lines. Content providers typically use such images to provide a visually pleasing line. However, these images can be simply transcoded using the HTML HR tag.
- **Icons** - images that are between 25x25 and 100x100 pixels in size are probably serving a role as icons. On typical computer displays (96 dpi monitors), this translates to between 0.25" and 1". These small images are typically used to give the user a hint of a larger image. Hence they can be transcoded to their HTML ALT text.
- **Banners** - images that are wider than 300 pixels with a height between 25 and 100 pixels are often used as banners. Typical advertisement banners and background images fall in this range. Even though advertisers might object to transcoding banners, a transcoding system can transcode banners with the HTML ALT text. We advocate that advertisers should provide network friendly ALT texts to advertise their products.
- **TrueImages** - We define images that do not belong to any of the categories mentioned above as TrueImages. Image level transcoding is typically appropriate for TrueImages. Obviously, some of our TrueImages may serve as icons or banners since we have been fairly conservative in our defining dimensions.

These image classification boundaries are illustrated in Figures 4 and 5.

We were concerned about mis-classifying a TrueImage as a bullet, line, banner or an icon. Hence, we performed an informal experiment to verify the validity of our classification scheme. We presented 275 random images from our image collection to a test subject (who did not know about our classification schemes' image geometry). We asked her to classify the images based on her notion of (advertisement) banners, (desktop) icons, lines and bullets. For our experiments, the images were displayed on a 75 dpi monitor. A comparison of our classification and the test subject's classification is tabulated in Table 1(a). We note about 21 TrueImages were mis-classified using our scheme described above as icons. We plot the dimensions of these mis-classified

Our imageclassification	Test subject classification				
	bullet	line	icon	banner	TrueImage
bullet	11	0	11	0	0
line	0	4	0	4	0
icon	0	0	23	0	21
banner	0	1	0	56	1
TrueImage	3	4	20	21	95

(a) Image category verification



(b) Dimensions of TrueImages that were mis-classified as icons

Table 1. Image category verification

Image Type	GIF images (%)	JPEG images (%)
Bullets	9.96	0.64
Lines	2.94	3.27
Icons	17.57	21.86
Banners	12.80	4.41
TrueImages	56.73	69.82

Table 2. Image category distribution

images in Figure 1(b). From Figure 1(b) we note most of these mis-classified images have a width greater than 80 pixels. A scheme that categorized icons as images between 25x25 and 75x75 pixels would have categorized these images as TrueImages. On our 75 dpi monitor, the notion of icons as objects less than 1” translates to a dimension of less than 75x75 pixels. These images should in fact have been classified as a TrueImage on this device. We note that the above image classification scheme seems to depend on the monitor resolution used in viewing the images. For further experiments, we use the image classification appropriate for 96 dpi monitors.

For our collection, the distribution of images in the various categories is tabulated in Table 2. From Table 2, we note that about 43% of GIF images and 30% of the JPEG images are for categories other than TrueImages. We note that TrueImages that are larger than 6 KB (throughout the rest of this paper, we will refer to TrueImages that are larger than 6 KB as LargeTrueImages) make up 181.3 MB (45.1%) and 344.5 MB (78.3%) of GIF and JPEG data, respectively. Hence, throughout the rest of this paper, we place special emphasis on the characteristics of LargeTrueImages.

To summarize, in this section, we analyzed our collection of images accessed in the Web for static characteristics such as their file size, geometry, number of unique colors and initial Quality Factor of JPEG images. We note that most of the GIF images (80%) are small (less than 6 KBytes). Simple transcoding filters such as bullets, icons, lines and banners remove about 45% of the GIF images from image transcoding consideration using simple HTML tags. We note that about 10% of the GIF images are LargeTrueImages. However, these images make up 45.1% of the data transferred for all GIF images. On the contrary, about 70% of the JPEG images are TrueImages. About 35% of the images are LargeTrueImages. These 35% of the images consume 78.3% of the JPEG data transferred. Hence, JPEG transcoding is promising for a larger set of images (35%) than for GIF images (10%) and also account for more of the consumed bandwidth.

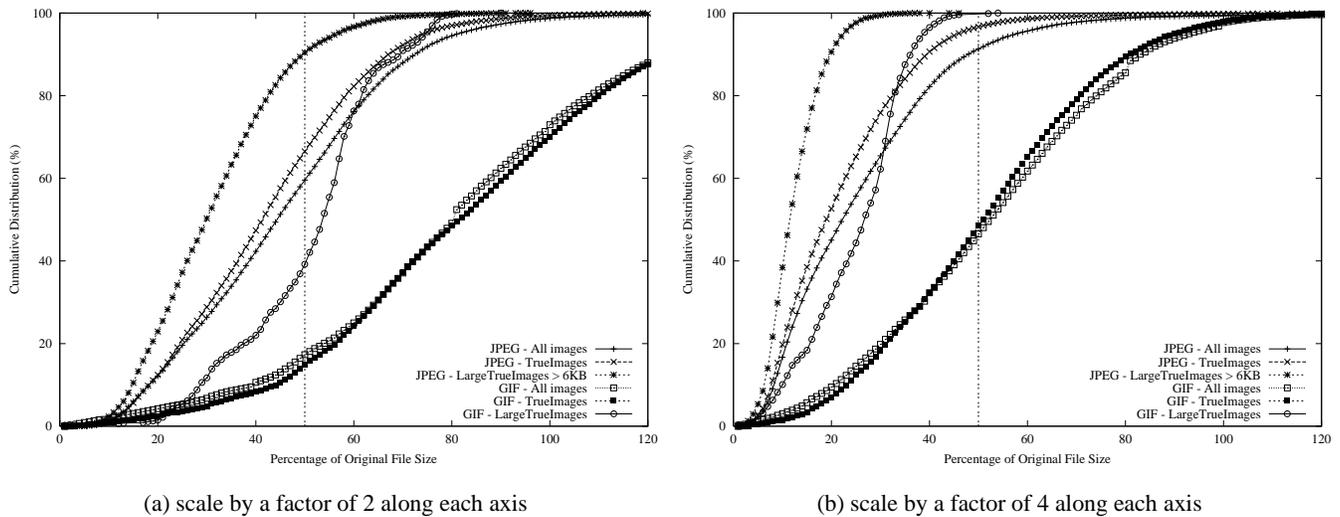


Figure 6. Transcoding that reduces the spatial geometry of images

4. IMAGE TRANSCODING CHARACTERISTICS

In the previous section, we analyzed the static characteristics of images accessed on the Internet. In this section, we analyze the transcoding characteristics of these images. For our study, we emphasize transcoding of LargeTrueImages. We explore transcodings that reduce the spatial geometry (frequently referred to as thumbnailing), the number of unique colors in an image, the JPEG compression metric as well as transcodings that change the image formats.

4.1. Transcoding that reduces the spatial geometry

First we analyze a transcoding that reduces the spatial size of an image, frequently referred to as thumbnailing. For our experiments, we reduce the spatial geometry of all the images as well as LargeTrueImages. We reduce the geometry by a factor of 2 and 4 along each axis (which translates to 25% and 6.25% of the original image pixels). This is achieved by using weighted averages of adjacent pixels. The resulting image file size as a percentage of the original image file size is plotted as a cumulative distribution in Figure 6(a) and Figure 6(b), respectively.

From Figure 6(a) we note that 65% of the JPEG images lose at least 50% of the image size for a transcoding that reduces the image geometry by a factor of 2 on each axis. Most (90%) of JPEG LargeTrueImages lose at least 50% of the image size. For GIF images however, 25% of the images actually increase in size compared to the original image (i.e., the image transcodes to a size that is more than 100% of the original image size). 20% of the GIF images save at least 50% of the original image file size. 40% of GIF LargeTrueImages save at least 50% of the original image file size.

Transcodings that reduce the spatial geometry by a factor of 4 along each axis are shown in Figure 6(b). We note that 40% of the GIF images and 90% of the JPEG images lose at least 50% of the image file size. We note that most (>99%) of the GIF and JPEG LargeTrueImages transcode to less than 50% of the original image file size.

A few GIF images (40% and 2% of the GIF images for a transcoding that reduces the image by a factor of 2 and 4 along each axis, respectively) transcode to a size that is larger than the original image file size (i.e., the image transcodes to a size that is more than 100% of the original image size). GIF^{18,19} uses a variation of the LZW compression algorithm to reduce the number of bits required to store frequently occurring color map values. In the GIF algorithm, pixels can be represented by 3 to 12 bits depending on the occurrence frequency of a particular color in the colormap. A transcoding that reduces the spatial geometry tends to increase the number of unique colors in an image as original color values are replaced by a new average color value. Since it takes more bits to represent less frequent pixels, introducing less frequent color values with low occurrence frequency leads to an increase in the output image size. This is against our goal for transcoding an image to reduce its size. We will use an example to illustrate this problem. One of the GIF images in our collection of geometry 345x145 (file size 2999 bytes and 7 unique colors) was transcoded to a GIF image of geometry 173x73. The new image was 3005 bytes and had 133 unique colors. The popular color (which will be represented by 3 bits) occurred 90.88% of the time in the original image,

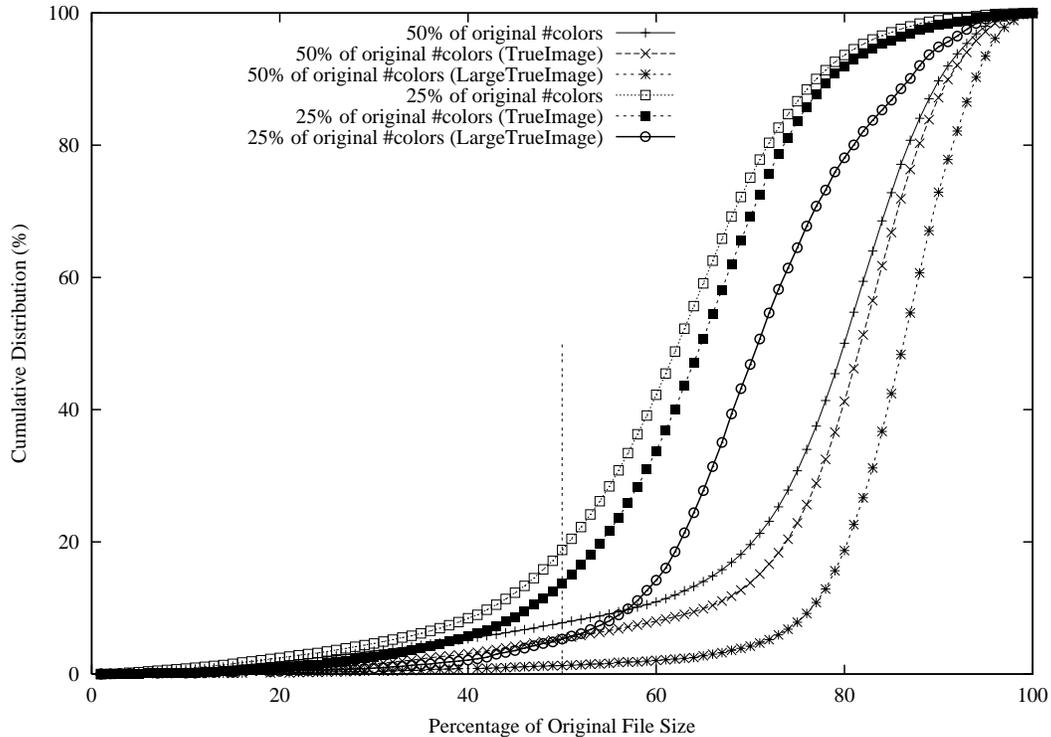


Figure 7. Transcoding that reduces the number of unique colors in GIF images

while it only occurred 85% of the time in the transcoded image. This is an illustration of how naive transcoding approaches can provide disappointing results due to subtle interactions. Subsequently reducing the number of unique colors in the transcoded image to 7 produced an output image size of 1276 bytes (even though this image was visually less pleasing). The images have to be transcoded to a sufficiently small spatial size to overcome this effect. This phenomenon can be observed by the increase in the number of images that show space saving between a transcoding that reduces the spatial geometry by a factor of 2 (60% in Figure 6(a)) and a transcoding that reduces the spatial geometry by a factor of 4 (98% in Figure 6(b)).

Reducing the image spatial geometry (both by a factor of 2 and 4 along each axis) is a productive transcoding for JPEG images as at least 50% of the images lose at least 50% of the file size. In the case of GIF images, reducing the image spatial geometry by a factor of 4 along each axis is a productive transcoding only for the GIF LargeTrueImages.

4.2. Transcoding that reduces the number of unique colors in an image

Next, we analyze transcoding operations that reduce the number of unique colors in an image.

For the GIF images in our collection, we reduce the number of unique colors for all the images as well as for LargeTrueImages. We reduce the number of unique colors by 1/2 and 1/4 of their original values. The resulting image file size as a percentage of the original image file size is plotted as a cumulative distribution in Figure 7.

From Figure 7 we note that for a transcoding that reduces the number of unique colors to 50% and 25% of the original unique colors, 8% and 20% of the images lose 50% of the original image file size, respectively. The results are worse for GIF LargeTrueImages, where for transcoding the images to lose 50% and 25% of the original unique colors, 2% and 5% of the images lose 50% of the original image file size, respectively. Reducing the number of unique colors tends to decrease the occurrence frequency of the most popular color by increasing the occurrence frequency of less frequent colors. This phenomenon of color value distribution change affecting the LZW compression was noted earlier in Section 4.1. Hence, reducing the number of unique colors in a GIF image is not a productive transcoding.

As discussed in Section 3.2, reducing the number of unique colors in a lossy compression such as JPEG is not appropriate. JPEG supports two color modes: GrayScale and TrueColor. We analyze the effectiveness of converting JPEG TrueColor images to GrayScale images for the JPEG images in our collection. The resulting image file size as a percentage of the original image file size is plotted as a cumulative distribution in Figure 8. From Figure 8, we note that 15% of the JPEG images lose 50% of

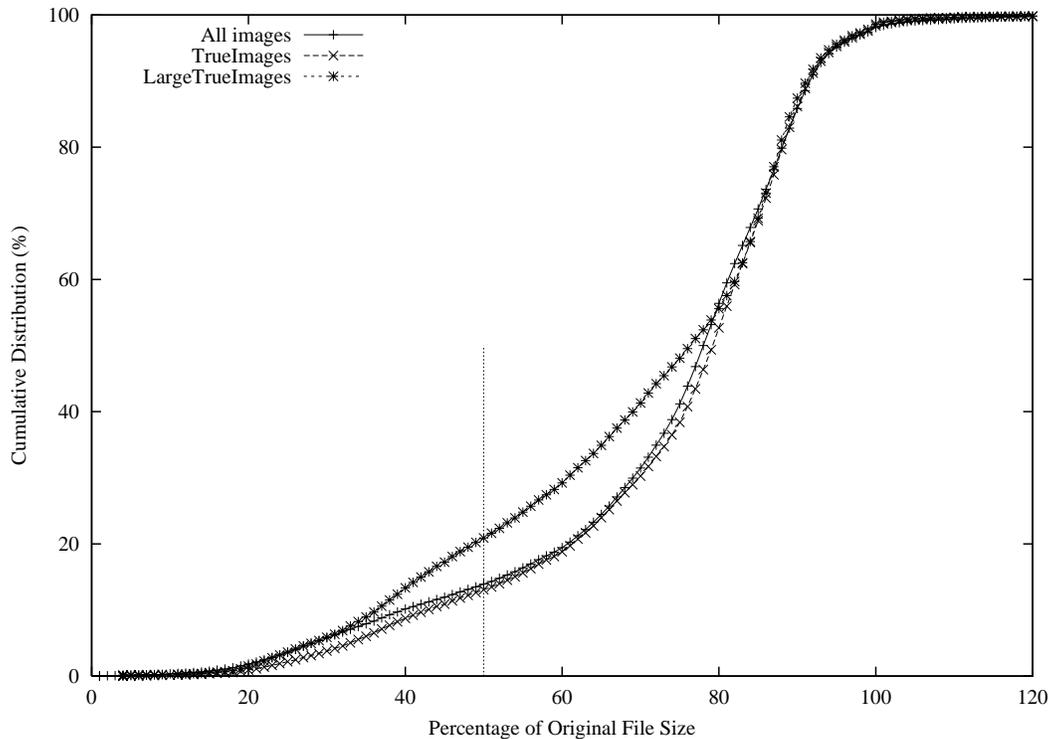


Figure 8. Transcoding that converts a TrueColor JPEG image to a GrayScale JPEG image

the image file size. LargeTrueImages perform slightly better than the entire image collection (25% of the images lose 50% of the image file size).

Since fewer than 25% of images lose 50% in image file size, reducing the number of colors does not appear to be a productive transcoding operation, either for GIF or JPEG images.

4.3. Transcoding that changes the image format

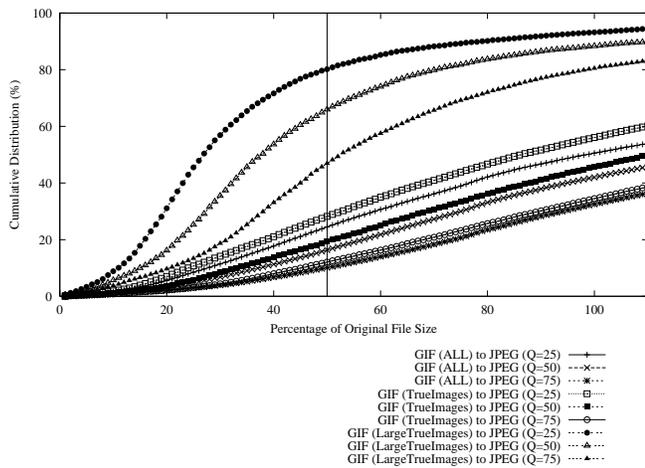
We now explore a transcoding that changes the format of the images themselves; GIF to JPEG and JPEG to GIF format. Even though JPEG compression is better suited for full color photographs and GIF format is better suited for line drawings, transcoding among the formats is a popular transcoding operation² and hence we explore the characteristics of this transcoding.

One variable in converting from a lossless compression technique such as GIF to a lossy compression algorithm such as JPEG is the choice of the compression metric for the JPEG images. For our experiments, we transcode the GIF images to JPEG images of Quality Factor values of 25, 50 and 75. The resulting image file size as a percentage of the original image file size is plotted as a cumulative distribution in Figure 9(a). The figure shows that transcoding GIF images to JPEG images with Quality Factor values of 25, 50 and 75 can lead to an increase in the image file size for 50%, 60% and 70% of the GIF images respectively. Since an increase in file size is against our goal for performing a transcoding, such a transcoding is not productive.

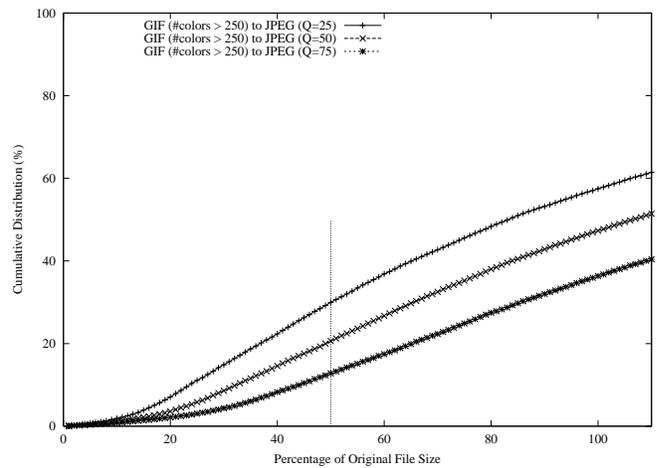
We note that transcoding to JPEG images of Quality Factor values of 25, 50 and 75 provide at least 50% saving in size for 50%, 65% and 80% of the GIF LargeTrueImages respectively. Hence transcoding GIF images to JPEG images is a productive transcoding for GIF LargeTrueImages.

GIF images are better optimized for line drawings with few colors. On the other hand, JPEG is optimized for photographs. Hence a GIF image with many colors might be expected to be better compressed as a JPEG image. To test this hypothesis, we analyzed the transcoding characteristics of transcoding GIF images with over 250 unique colors to JPEG images. The resulting image file size as a percentage of the original image file size is plotted as a cumulative distribution in Figure 9(b)

From Figure 9(b), we note that such a transcoding is not productive as less than 30% of the images provide savings of 50% of the image file size. In order to better understand this counter-intuitive result, we visually inspected these GIF images and found that most of these GIF images appear to be computer generated (by composing photographs and textual information).



(a) GIF to JPEG images



(b) GIF (#colors > 250) to JPEG images

Figure 9. Transcoding that converts GIF images to JPEG images

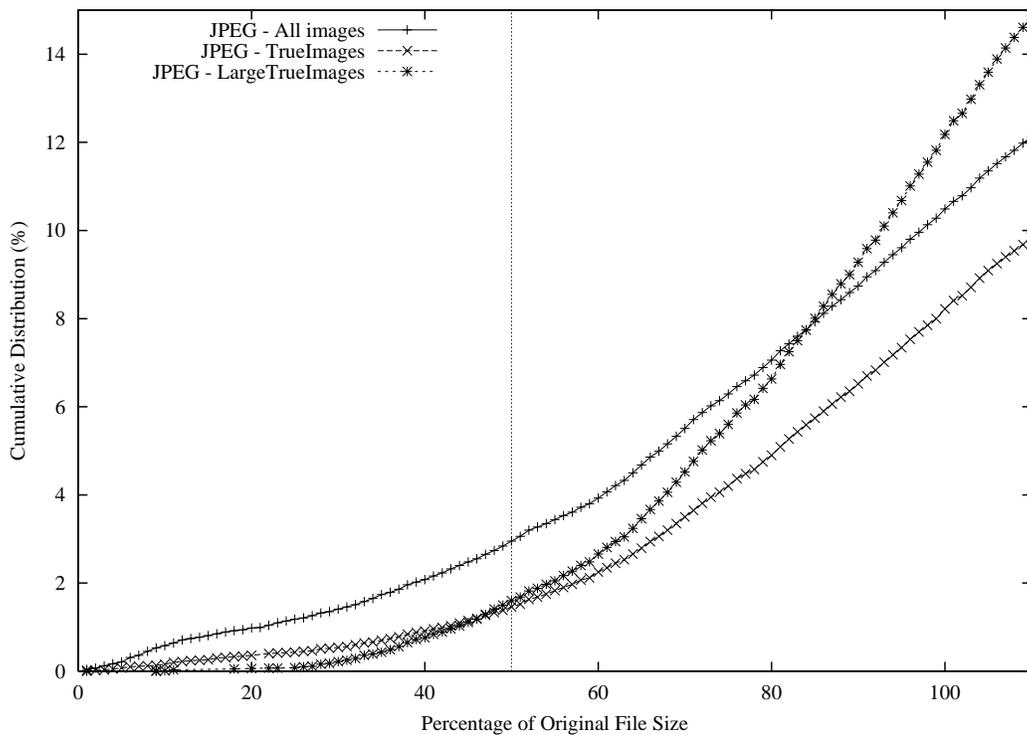


Figure 10. Transcoding that converts JPEG images to GIF images (notice change of scale of Y axis)

Hence, even though the number of unique colors is over 250, the images are still overwhelmingly line drawings and hence are better encoded as GIF images.

Next we analyze the transcoding from JPEG to GIF images. Conversion from JPEG to GIF is via the spatial domain, which is straightforward. The resulting image file size as a percentage of the original image file size is plotted as a cumulative distribution in Figure 10 (notice change of scale of Y axis). From Figure 10, we note that transcoding JPEG to GIF images is not a productive transcoding as less than 5% of the images show savings of 50% of the original image file size. Most images (90%) are larger than the original JPEG images.

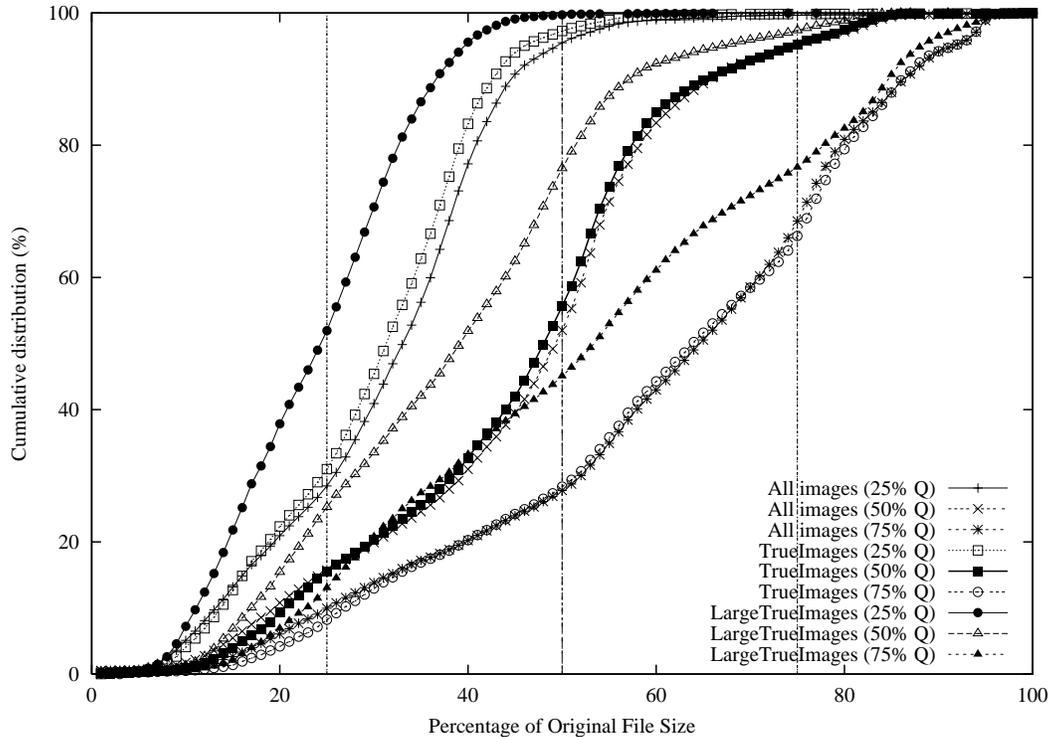


Figure 11. Transcoding that reduces the JPEG image Quality Factor

In summary, converting GIF images to JPEG images is a productive transcoding only for GIF LargeTrueImages as at least 50% of the images save at least 50% in file size. We recognize there may be other reasons for such conversions (e.g., to match the capabilities of the client), but reliable size reduction is not an appropriate justification for most images.

4.4. Transcoding that changes the JPEG Compression Metric

Next we explore transcodings that change the level of “lossiness” in an image. Lossy compression techniques such as JPEG define a way to control the amount of lossiness using a compression metric. The compression metric utilizes quantization tables such that images lose least perceptible artifacts first. Hence, changing the compression metric is intuitively a good transcoding metric for JPEG images. Earlier work²⁸⁻³⁰ confirmed that JPEG compression metrics such as JPEG Quality Factor outperformed other measures in predicting the quality loss of an image.

For our experiments, we reduce the image Quality Factors to 25%, 50% and 75% of the original image Quality Factor values. The resulting image file size as a percentage of the original image file size is plotted as a cumulative distribution in Figure 11. From Figure 11, we note that cumulative distribution of the file sizes for all the images are similar to the distribution for JPEG TrueImages. For a transcoding that reduces the image Quality Factor to 25%, 95% of all the JPEG images lose 50% of the file size while over 99% of all the JPEG LargeTrueImages lose 50% of the file size. For a transcoding that reduces the image Quality Factor value to 50%, we note that 50% of all the JPEG images lose 50% of file size and 80% of all the JPEG LargeTrueImages lose 50% of file size. Finally, for a transcoding that reduces the image Quality Factor value to 75% we note that 25% of all the JPEG images lose at least 50% of file size and 45% of all the JPEG LargeTrueImages lose 50% of file size. Hence, we conclude that changing the JPEG compression metric is a productive image transcoding.

Next, we utilize the notion of efficient transcoding that we developed earlier²⁷ where we defined efficient transcoding as a transcoding that loses more in file size than image Quality Factor. An efficient transcoding gives a measure of worthwhile savings in size. Since for our experiments we reduce the image Quality factors to 25%, 50% and 75% of the original image Quality Factor values, an efficient transcoding would lose at least 25%, 50% and 75% in size respectively. In Figure 11, we draw the threshold 25%, 50% and 75% lines to mark the efficient images for the various transcodings. From Figure 11, we note that for transcodings that transcode images to 25%, 50% and 75% of the initial Quality Factor values, 25%, 50% and 70% of the JPEG images transcoded efficiently. We also note that 50%, 75% and 75% of LargeTrueImages transcoded efficiently for the same transcodings

In summary, we note that the JPEG compression metric is a productive transcoding operation for all JPEG images. In addition, for JPEG LargeTrueImages, reducing the spatial geometry (thumbnailing) is a productive transcoding. On the other hand, the only productive transcodings for GIF images are for transcoding GIF LargeTrueImages; transcoding to JPEG images and a transcoding that reduces the spatial geometry at least by a factor of 4 along each axis are productive. None of the other image transcoding techniques explored provide significant savings for all GIF images.

5. RELATED WORK

5.1. Web object characterization

Bray³¹ and Woodruff³² analyzed the object characteristics of HTML documents. They examined the document size, number and types of tags, attributes, file extensions, protocols, ports, the number of in-links and the ratio of document size to the number of tags and attributes. Acharya et al.³³ analyzed the characteristics of video objects stored on the Internet. Acharya et al.³⁴ also analyzed how users access the video objects. Our work presents a similar analysis of internet images.

5.2. Image Transcoding

A number of research and commercial products have used various forms of image transcodings to customize an image for the prevailing network conditions or the destination target device characteristics. Fox et al.² used transcoding to render an image on a PDA such as Palmpilot, as well as to offset access latencies from slow modems. Noble et al.⁴ manipulated the JPEG Compression metric as a distillation technique for a Web browser that adapts to changing network environments. Mazer et al.⁵ describe a framework for allowing users to specify their own transcoding transducers for a application-specific proxy that acts on the HTTP stream to customize the stream for a particular client. Ortega et al.⁹ have used JPEG progressive encoding to recode images to lower resolutions, thereby increasing the effective cache size. Commercial products such as WebExpress¹⁰ and Websphere¹⁵ from IBM, Portal-To-Go¹⁴ from Oracle, QuickWeb technology¹¹ from Intel, Fastlane¹³ from Spectrum Information technology and Johnson Grace ART format¹² from AOL have used various forms of compression and transcoding operations to improve Web access from slow networks.

Even though transcoding has been widely used in a number of systems to deal with network access latencies, display characteristics or storage space requirements, there has been little formal work in conducting a systematic study to measure the information loss associated with a given transcoding. Han et al.^{35,36} present an analytical framework to quantify transcoding trade-offs.

In our earlier work,²⁷ we developed techniques to measure the initial information Quality Factor of a JPEG image as well as the computational costs and space benefits for a transcoding that changes the JPEG compression metric of an image. We utilized this informed transcoding technology in a transcoding Web proxy serving mobile clients that access the Web using slow and expensive networks.¹ We demonstrated that it is possible to use informed transcoding techniques to balance the demand for quality of multimedia content while reducing consumed network bandwidth and server CPU overhead. We also utilized informed transcoding in a Web service that offers differentiated Quality of Service for different client classes.¹⁷ We demonstrated that it is possible to use informed transcoding techniques to provide differentiated service and to dynamically allocate available bandwidth among different client classes, while delivering a high degree of information content (quality factor) for all clients.

6. CONCLUSIONS

This paper makes the following contributions to our understanding of the characteristics of images accessed on the Web and the use of image transcoding applied to them for size reductions:

- Most of the GIF images are small; about 80% of the GIF images are smaller than 6 KBs. Transcoding for size reduction is hardly worthwhile for such small images.
- About 45% of the GIF images can be characterized as typical of bullets, icons, lines or banners. Simple HTML tag transcoding such as HR or LI is more effectual for such GIF images.
- About 15% of the GIF images are animated GIF's. An obvious way to deal with large GIF files composed of numerous animations is to reduce the number of individual animation frames.
- Approximately 10% of the GIF images are LargeTrueImages. However they make up 45.1% of the GIF data transferred.

- For GIF LargeTrueImages, transcodings that reduce the spatial geometry by at least a factor of 4 along each axis as well transcoding the GIF images to JPEG images are productive (where at least 50% of the images lose at least 50% of image file size).
- Popular transcoding techniques such as thumbnailing of GIF images have the potential of actually increasing the output image size.
- There are more transcoding opportunities for JPEG images. On average JPEG images are larger than GIF images. 40% of the JPEG images are larger than 6 KBs. About 30% of JPEG images are LargeTrueImages as we define them. 78.3% of the JPEG data is transferred for LargeTrueImages (35.6%).
- For JPEG TrueImages, the JPEG compression metric and a transcoding that reduces the spatial geometry are productive transcodings. Since the compression metric loses visually imperceptible information first, it is a better transcoding since it loses image file size while sacrificing as little visual information as possible.

We are currently investigating the transcoding issues associated with the interlaced GIF and progressive JPEG formats. These formats allow clients to terminate an streaming image after the required amount of image data is downloaded.

REFERENCES

1. S. Chandra, C. S. Ellis, and A. Vahdat, "Multimedia Web Services for Mobile Clients Using Quality Aware Transcoding," in *The Second ACM International Workshop on Wireless Mobile Multimedia*, pp. 99–108, (Seattle), Aug. 1999.
2. A. Fox and E. A. Brewer, "Reducing www latency and bandwidth requirements via real-time distillation," in *Proc. of Fifth International World Wide Web Conference*, pp. 1445–1456, (Paris, France), May 1996.
3. A. Fox, S. D. Gribble, E. A. Brewer, and E. Amir, "Adapting to network and client variability via on-demand dynamic distillation," *ACM SIGPLAN Notices* **31**, pp. 160–170, Sept. 1996.
4. B. D. Noble, M. Satyanarayanan, D. Narayanan, J. E. Tilton, J. Flinn, and K. R. Walker, "Application-aware adaptation for mobility," in *Proc. of the 16th ACM Symposium on Operating Systems and Principles*, (Saint-Malo, France), Oct. 1997.
5. M. S. Mazer, C. Brooks, J. LoVerso, L. Theran, F. Hirsch, S. Macrakis, S. Shapiro, and D. Rockwell, "Distributed clients for enhanced usability, reliability, and adaptability in accessing the national information environment," tech. rep., The Open Group Research Institute, 11 Cambridge Center, Cambridge MA 02142, 1998.
6. M. Liljeberg, H. Helin, M. Kojo, and K. Raatikainen, "Mowgli www software: Improved usability of www in mobile wan environments," in *IEEE Global Internet 1996*, IEEE Communications Society, (London, England), Nov. 1996.
7. M. Liljeberg, T. Alanko, M. Kojo, H. Laamanen, and K. Raatikainen, "Optimizing world-wide web for weakly connected mobile workstations: An indirect approach," in *Proc. of 2nd International Workshop on Services in Distributed and Networked Environments (SDNE'95)*, (Whistler, Canada), June 1995.
8. J. Kangasharju, Y. Kwon, and A. Ortega, "Design and implementation of a soft caching proxy," in *3rd Intl. WWW Caching Workshop*, (Manchester, England), June 1998.
9. A. Ortega, F. Carignano, S. Ayer, and M. Vetterli, "Soft caching: Web cache management techniques for images," in *IEEE Signal Processing Society 1997 Workshop on Multimedia Signal Processing*, (Princeton NJ), June 1997.
10. R. Floyd, B. Housel, and C. Tait, "Mobile Web Access using eNetwork Web Express," *IEEE Personal Communications* **5**, Oct. 1998.
11. "Intel quickweb." <http://www-us-east.intel.com/quickweb/>.
12. America Online Inc., "Johnson grace ART image format."
13. Spectrum Information Technologies Inc., "Fastlane." <http://www.spectruminfo.com/>.
14. Oracle Inc., "Oracle portal-to-go; any service to any device," tech. rep., Oracle Inc., <http://www.oracle.com/mobile/portaltogo/>, Oct. 1999. Oracle business white paper.
15. IBM, "Websphere transcoding publisher." www.ibm.com/software/webservers/transcoding/, 2000.
16. J. R. Smith, R. Mohan, and C.-S. Li, "Scalable multimedia delivery for pervasive computing," in *Proc. of ACM Multimedia '99*, pp. 131–140, ACM Multimedia, (Orlando, Florida), Oct. 1999.
17. S. Chandra, C. S. Ellis, and A. Vahdat, "Differentiated multimedia web services using quality aware transcoding," in *Nineteenth Annual Joint Conference Of The IEEE Computer And Communications Societies (INFOCOM)*, (Tel Aviv, Israel), Mar. 2000.
18. CompuServe Inc., 5000, Arlington Centre Blvd, Columbus, OH 43220, *Graphics Interchange Format (GIF) - A standard defining a mechanism for the storage and transmission of raster-based graphics information*, June 1987.

19. CompuServe Inc., 5000, Arlington Centre Blvd, Columbus, OH 43220, *Graphics Interchange Format - Version 89a*, 1989.
20. E. Hamilton, *JPEG File Interchange Format - Version 1.02*. C-Cube Microsystems, Sept. 1992.
21. J. Mogul and T. M. Kroeger, "Digital's web proxy traces." <ftp.digital.com/pub/DEC/traces/proxy/webtraces.html>, 1996.
22. P. Danzig, J. Mogul, V. Paxson, and M. Schwartz, "Internet traffic archive." <http://ita.ee.lbl.gov/>.
23. The National Laboratory for Applied Network Research, "A distributed testbed for national information provisioning." <http://ircache.nlanr.net/>.
24. T. Hansen and J. Poskanzer, "Netpbm release 7."
25. T. Lane, P. Gladstone, L. Ortiz, J. Boucher, L. Crocker, J. Minguillon, G. Phillips, D. Rossi, and G. Weijers, "The independent jpeg group's jpeg software release 6b." <ftp://ftp.uu.net/graphics/jpeg/jpegsrc.v6b.tar.gz>.
26. J. Nielsen, *Usability Engineering*, Academic Press, Boston, MA, 1993. (hardcover), 0-12-518406-9 (paperback).
27. S. Chandra and C. S. Ellis, "JPEG Compression Metric as a Quality Aware Image Transcoding," in *Proc. of 2nd Symposium on Internet Technologies and Systems*, USENIX, (Boulder, CO), Oct. 1999.
28. A. M. Ford, *Relations between Image Quality and Still Image Compression*. PhD thesis, Univ. of Westminster, May 1997.
29. R. E. Jacobson, A. M. Ford, and G. G. Attridge, "Evaluation of the effects of compression on the quality of images on a soft display," in *Proc. of SPIE: Human Vision and Electronic Imaging II*, (San Jose, CA), Feb. 1997.
30. A. M. Ford, R. E. Jacobson, G. G. Attridge, and M. R. Pointer, "Effects of image compression on image quality," in *Royal Photographic Society Symposium on Photographic Image Quality*, (London), Sept. 1994.
31. T. Bray, "Measuring the web," in *Proc. of the Fifth International World Wide Web Conference*, pp. 994–1005, (Paris, France), May 1996. http://www5conf.inria.fr/fich_html/papers/P9/Overview.html.
32. A. Woodruff, P. M. Aoki, E. Brewer, P. Gauthier, and L. A. Rowe, "An investigation of documents from the world wide web," in *Proc. of the Fifth International World Wide Web Conference*, pp. 963–979, (Paris, France), May 1996. http://www5conf.inria.fr/fich_html/papers/P7/Overview.html.
33. S. Acharya and B. Smith, "An experiment to characterize videos stored on the web," in *Multimedia Computing and Networking (MMCN'98)*, K. Jeffay, D. D. Kandlur, and T. Roscoe, eds., vol. 3310, pp. 166–178, SPIE and IS&T, (San Jose, CA), Jan. 1998.
34. S. Acharya, B. Smith, and P. Parnes, "Characterizing user access to videos on the world wide web," in *Multimedia Computing and Networking (MMCN'00)*, K. Nahrstedt and W. Feng, eds., vol. 3969, SPIE and IS&T, (San Jose, CA), Jan. 2000.
35. R. Han, P. Bhagwat, R. LaMaire, T. Mummert, V. Perret, and J. Rubas, "Dynamic adaptation in an image transcoding proxy for mobile web browsing," *IEEE Personal Communications Magazine* **5**, pp. 8–17, Dec. 1998.
36. R. Han, "Factoring a mobile client's effective processing speed into the image transcoding decision," in *The Second ACM International Workshop on Wireless Mobile Multimedia*, pp. 91–98, (Seattle), Aug. 1999.