1996

A Color-based Technique for Measuring Visible Loss for Use in Image Data Communication

Melliyal Annamalai

Aurobindo Sundaram

Bharat Bhargava
Purdue University, bb@cs.purdue.edu

Report Number:
96-029

A COLOR-BASED TECHNIQUE FOR MEASURING VISIBLE LOSS FOR USE IN IMAGE DATA COMMUNICATION

Melliyal Annamalai
Aurobindo Sundaram
Bharat Bhargava

Department of Computer Sciences
Purdue University
West Lafayette, IN 47907

CSD TR-96-029
May 1996

A Color-based Technique for Measuring Visible Loss for Use in Image Data Communication [1]

Melliyal Annamalai, Aurobindo Sundaram and Bharat Bhargava
Department of Computer Sciences
Purdue University
W. Lafayette, IN 47906, USA
{melli,auro,bb}@cs.purdue.edu

Abstract

The concept of the global information infrastructure and specifically that of the World Wide Web (WWW) has led to users accessing data of different media including images and video data over a wide area network. These data objects have sizes the order of megabytes and communication time is very large. The data size can be reduced without losing information by applying loss-inducing techniques and this will lead to reduction in communication time. Several loss-inducing techniques have been developed and each image is treated differently by each technique. In some cases an acceptable quality of the image is obtained and in some cases it is not. In this paper we develop a color-based technique to quantify the data loss when a loss-inducing technique is applied to an image. This will result in estimating whether the resulting image is indistinguishable from the original *with respect to the human eye*. We illustrate its use to classify images according to the loss they can tolerate. This avoids redundant communication of a high quality image when a lower quality image can satisfy the application resulting in the conservation and better usage of network resources. We present the technique, the communication time saved, and an experimental evaluation to prove the validity of the technique.

1 Introduction

The Global Information Infrastructure brings about the integration, management, and communication of gigabytes of data in a parallel and distributed environment over national and international networks [3]. A global digital library should provide everyone in the world access to distributed stores of information at a reasonable cost. Digital library data objects are distributed; they are large in size, of the order of gigabytes. The inherently distributed nature of digital libraries results in communication being a bottleneck in the current network technology scenario. Researchers are developing faster network technologies like ATM, FDDI, DQDB etc., but the increasing number of users ensures that the communication bottleneck is going to be a problem in the foreseeable future.

Image and video data are part of the global digital library and are important because of the growing emphasis on visual information systems. Some examples of applications dealing with images are remote sensing, medical imaging, and geographical information systems. Some examples of applications dealing with video are video-on-demand systems and video-conferencing.

[1]This research is supported by NSF under grant number NCR-9405931 and a NASA fellowship

These applications require the communication of image and video data in a distributed environment. Image data is large in size ranging anywhere from 200K to 20000K. Video data can be the order of a few megabytes and each frame can be several hundred kilobytes. This large size leads to increased processing time during search, retrieval and other operations, especially during communication of the data. Lossy compression, resolution reduction, interlacing, and lossy transmission are some of the techniques that have been used to reduce the size of the image or video frame. The result of these techniques is the reduction of communication time and storage space. Visual data like images and video are rich in semantic content and careful manipulation will result in data so that no visible information is lost [1]. They can be stored, manipulated and viewed in multiple levels of quality. But how does one evaluate exactly how much information has been lost ? How does one decide that no "visible" information has been lost ? How does one determine that data might be lost, but it "looks" the same ? There is a need to quantify the amount of information that is lost. This will enable us to define quality levels for an image. Such a quantification is useful because:

1. We can determine the best quality image that can be transmitted using the available resources
2. Quantifying the loss is the first step towards minimizing this loss while transmitting images

In this paper we attempt to develop a technique to perform this quantification. Our technique can be used in any situation involving the communication of a set of color pixels representing visual data. It can be applied to images or to frames in a video-conferencing application. For our purpose an image can be considered equivalent to a video frame since they both are two-dimensional data structures containing a set of color pixels. In the rest of this paper we use the word image to refer to both images and video frames.

1.1 Motivation

Our research in multimedia communication has led to observations which indicate that a reduction in response time can be obtained by reducing the amount of data to be transmitted with no visible information loss. Lossy compression techniques can result in an image which has less data and lower quality. For several applications it would suffice to get a lower quality image if the response time is lower [1]. School children accessing NASA's repository of image data is one such application. In such a scenario, a lower quality image which is visually indistinguishable from the original is sufficient. Another application is video-conferencing which has a realtime requirement of maintaining frame rate. A lower quality frame would suffice as long as the video-conferencing session is not disrupted [4] and the frame does not completely deteriorate in quality.

Current techniques of communicating multimedia data over a wide area network function by transmitting *all* the data. We argue that for some applications this is unnecessary and that bandwidth usage can be reduced by communicating only as much data as required by the quality level of that application.

Our motivation behind this work was to quantify the amount of data lost with respect to the human eye. Such a quantification would help in deciding when data can be lost and when it cannot be lost. We would like to classify images according to the amount of data they can lose. Different loss-inducing techniques operate on different images differently. No technique has an uniform method of operating on all images. For example, it is known that GIF performs better than JPEG when uniform color segments are involved but JPEG performs better than GIF for natural scene images [6]. Every technique exploits image content, or image properties like texture, color, repeated patterns differently to result in a lower image size. There is a need to quantify data loss so that the images can be classified whatever the techniques used for inducing loss.

1.2 Applications of the Technique

We have identified areas of multimedia communication where such a technique will be useful:

- Defining the various quality levels in an image: This will help in transmitting only exactly the amount of data required to achieve acceptable quality and will hence reduce bandwidth usage. A static analysis of the images in a data repository can be performed to find out the different levels of quality for that image.
- In a video-conferencing environment, the technique can be used to provide a computationally efficient method of evaluating the quality of frame to be transmitted in realtime, once again resulting in reduced bandwidth usage. In an environment where frames are being transmitted and received at the rate of 30 frames/second, frame quality can be reduced to a level where the human eye cannot distinguish the difference.

2 Quality Levels of Images

As we have discussed different images require different amounts of processing, or lossy techniques applied to them, to reach the same quality level. When information is lost in an image, it is regenerated based on existing information during display. This could be based on the surrounding pixels as in resolution reduction or the high frequency DCT components as in JPEG compression and so on. How good this regeneration is varies from image to image and depends on the semantics of the image. The amount of "lossiness" an image can tolerate depends on several factors:

- **Total number of colors**: First of all, if an image contains few colors then there is the possibility of a high degree of compression due to run length encoding and similar techniques. For instance, GIF format divides an image into blocks and when a block similar to an earlier block is encountered, the block is replaced by a pointer to the previous block. Thus many similar patterns implies a high degree of compression and fewer the colors, more the probability of finding a pattern. Secondly, fewer colors implies that in the case of loss, it can be regenerated accurately using the existing information. This is because the existing information will be very similar to the data that is lost and any simple interpolation technique will suffice. For example, consider the image of a green square on a red background. Any pixels within the green square or in the background can be regenerated with 100% accuracy. Only the pixels on the border of the green square might appear blurred and different from the original.

- **Total number of distinct colors in the image:** (A 'distinct' color is one related to another color in that image by being a shade of that color.) For instance, a shade of red would be indistinguishable from red. If an image contained spots of a shade of red surrounded by red, then the loss of pixels of the color a shade of red might be regenerated by using the red color. This will lead to an image indistinguishable from the original.
- **The organization of colors in the image**: Regeneration often uses as input information from the neighborhood of the area of loss. Thus the blending of colors into one another will lead to a better regenerated image than an image containing a sharp outline between two distinct colors.

We define three quality levels:

1. **Q1**: Perfect, no change from the original: This quality is useful for scientists and engineers who will be interested in conducting further automated processing of the data received. Loss would be intolerable in such cases.
2. **Q2**: Indistinguishable from the original with respect to the human eye: This quality is useful for school children in a virtual classroom session on geography, users interested in browsing through the data repository, and so on. The image will be looked at for only a few minutes and as long as the image is clear to the human eye, the quality is sufficient.
3. **Q3**: Blurred, but user can identify objects: This quality is useful for quick browsing applications. The loss is noticeable to the human eye but the contents and information contained in the image are clearly perceivable.

3 Description of the Technique

We have developed a technique based on the color histogram of an image to evaluate the quality of an image. Color is one of the criteria used by human eyes to identify objects around them. A color histogram of an image is a series of bins representing the different colors in the image. Each bin contains the number of pixels in the image that have that color. For example, let us consider a simple image with three colors red, green, and blue, and resolution 3x3 (totally nine pixels). A color histogram for that image might be:

```
red: 4     blue: 2     green: 3
```

Color histograms are invariant to slight modifications of position and scaling and hence provide a more accurate measure than pixel-to-pixel matching. Color histograms have been used in several prototype systems for indexing and retrieval of images [5, 8, 10]. They have also been used for automatic segmentation of video data into distinctive scenes to enable retrieval later on [2].

The technique we have proposed takes as input an image and its loss-induced version and outputs the *col_diff* (color difference) between them. The color difference is computed by evaluating the difference between the corresponding bins of the two color histograms.

3.1 Methodology

The following is a sequence of steps which constitute the methodology:

Input: Image X and its loss-induced version L. n is the total number of colors in X.

1. Compute color histogram for image X, $hist^A$.
2. Compute color histogram for image L, $hist^L$.
3. Quantize both histograms to merge colors which are similar to each other.
4. Evaluate *col_diff* as:

$$col\_diff = \frac{\sqrt{\sum_{i=1}^{n}(hist_i^X - hist_i^L)^2}}{\sum_{i=1}^{n} hist_i^L * n}$$

5. Compare *col_diff* with the specified *thresh*.

The value *thresh* is described in the next subsection.

3.2 Selection of the value *thresh*

Each quality level has a *thresh* value associated with it. An image should not lose more data than the associated *thresh* value. The *thresh* value thus decides the amount of lossy compression or other lossy techniques that should be applied. Different images need different amounts of compression to satisfy the same *thresh* value. Depending on the requirements of the application, an appropriate quality level can be chosen instead of quality level Q1 being transmitted all the time. This leads to a usage of resources only when absolutely necessary. An image database server should contain a module to decide whether the loss-induced image satisfies the quality required by the application (or requested by the user) or not. The resulting saving in communication time can be as much as 75% depending on the distance the image should be transmitted. This results in a lower processing time even allowing for the extra processing time required in the case of dynamically determining the quality level of an image. If the quality levels for the image are precomputed then the extra processing time is also saved. Table 1 presents the communication times for the original image and the loss-induced image (the loss inducing technique here is compressing to 10% JPEG). In this example image data size is 480K. The evaluation of communication time was presented in an earlier paper [3].

3.3 Granularity

Granularity refers to the number of colors used to form the histogram and calculate *col_diff*. Different file formats use different number of colors. GIF uses only 256 colors (8-bit) while JPEG uses 16 million colors (24-bit). Quantization refers to the step where the number of colors are reduced. Colors which are shades of the same color are merged into one color. This is to merge colors which are not distinguishable as different by the human eye. Quantization functions by smoothing over differences. For example, consider the two colors represented by

Remote Site in	Round Trip Time for Original Image (ms)	Round Trip Time Loss-induced Image (ms)
Illinois (19 hops away)	17920.500	1899.100
New York (19 hops away)	74076.102	4588.400
Maryland (22 hops away)	69887.383	7907.300
Texas (23 hops away)	68397.336	4322.700
California (25 hops away)	95421.445	5717.100

Table 1: Comparison of Communication Times of Original and Loss-induced Images (Image Data Size is 480K)

the (r,g,b) triples (25,25,26) and (25,26,27). These are visually the same and can be merged to represent a single color. We quantized the total number of colors to be 256 in all cases. Our experiments showed us that this was sufficient to measure the distortion in the image. If all 16 million colors were used then the difference colors added up to result in a high *col_diff* which was not indicative of the human eye response to the distortion in the image. Using 256 colors captured the distortions visible to the human eye.

4 Experimental Evaluation

4.1 Application of the Technique to Lossy Methods

We refer to an operation which when applied to an image results in a lower quality (and consequently smaller size) image as a *lossy* operation. We applied our technique to two lossy methods and evaluated the results by computing the *col_diff* values in both cases and comparing them with the opinions of users. The element of bias in the user evaluation is unavoidable because of the nature of the experiment.

4.2 Experimental Infrastructure

We conducted our experiments using a Sparc 5 and Sparc 10 (running Solaris 4.3) at the RAID (robust, adaptable, interoperable, distributed, database) laboratory. We used 150 image files downloaded from NASA and other sites on the World Wide Web. The images were chosen to represent natural scenes and photographs, images focusing on an individual object and groups of objects, images with a multi-hued background and images with a plain background.

4.2.1 Experiment 1: Resolution Reduction

One way of reducing the size of an image is to reduce its resolution. Representing four pixels by one pixel results in an image which is one fourth the size of the original. The original image can be reconstructed by expanding each pixel in the smaller image into four pixels (see figure 1).

$$X = \frac{X1+X2+X3+X4}{4}$$

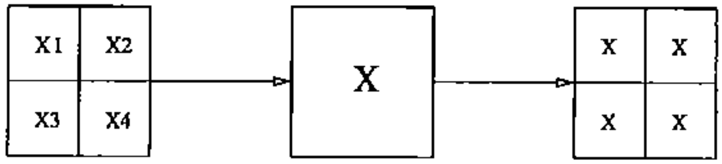

Figure 1: Illustration of Resolution Reduction

Problem Statement: The purpose of this experiment was to verify the validity of our technique using the loss-inducing technique of resolution reduction. Resolution reduction results in some loss of information. For some images, this loss is indistinguishable while for others, it is. We were interested in comparing the loss as perceived by users with the loss as calculated using our technique.

Method: We reduced the resolution of the 150 images by one-half and one-quarter. Users were asked to compare both versions with the original and evaluate their quality on a scale of one to ten (one was excellent and ten was bad). The difference in quality was also evaluated using our technique and the resulting *col_diff* value compared with the user given rank. Table 2 lists ten images with their *col_diff* values for half and quarter resolutions and the rank given by users for half resolution images. For convenience, the *col_diff* values are scaled by 10^7.

Name	Half Res.	Quart. Res.	User Rank (for Half Res.)
Lion	2.78	2.94	1
Flowers	6.71	7.98	1
Scenery	4.69	5.38	1
Asteroid	11.87	13.04	2
Panda	7.65	17.45	2
Flower	12.04	30.08	2
Curve	43.64	66.13	4
Mars	34.38	80.63	4
Galaxy	123.55	120.43	7
Red	151.77	182.78	7

Table 2: *Col_diff* Values when Image Resolutions are Reduced to One-half and One-fourth

Discussion: We can observe from the Table 2 that when the calculated *col_diff* value is low the user perceived quality is very good, and when the *col_diff* value is high the user perceived quality is bad. In the case of some images like Red, the high value for *col_diff* is due to the prescence of text in the image. The text is completely distorted when the image resolution is reduced by half making the quality unacceptable. In the case of Galaxy, there is very little color other than the background (black), but that color is completely distorted. This fact is reflected by the high *col_diff* too. Figures 2 and 3 show the original and half resolution versions for Lion

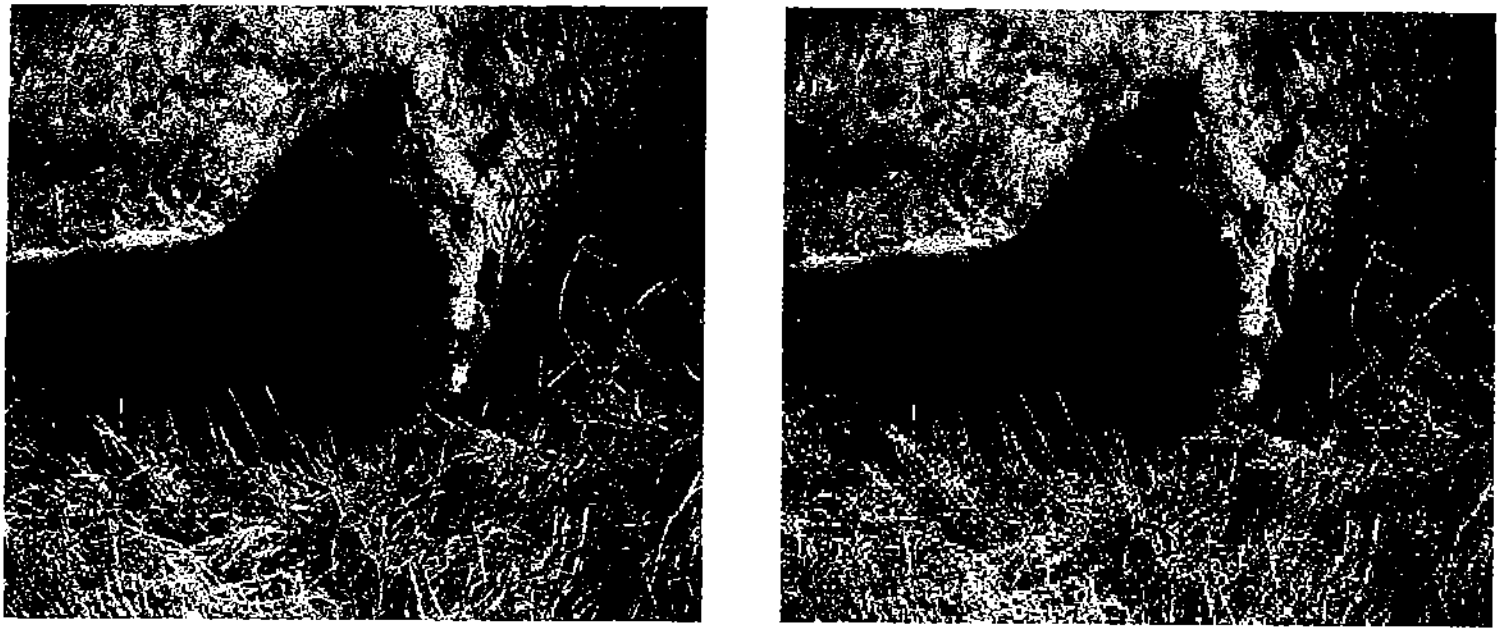

Figure 2: Original and Half Resolution Versions of the Lion Image

and Red images respectively. It can be observed that the Lion image with half resolution is indistinguishable from the original while the Red image with half resolution is distinguishable. This is mainly due to the prescence of text in the Red image.

4.2.2 Experiment 2: Lossy Compression

Lossy compression is another loss-inducing technique. We chose JPEG as our lossy compression technique [11]. It does not reconstruct the original image bit-for-bit but reconstructs an image which to the human eye looks very similar to the original. JPEG primarily stores information on color changes, particularly variations in brightness, because the eye is very sensitive to this [7]. One can choose the extent of compression while choosing JPEG. The extent of compression decides how much data is lost. We use percentages to denote the compression level. A 10% JPEG file is one which has retained 10% of the original file. This does not necessarily mean that the file size is 10% of the original size.

Problem Statement: The purpose of this experiment was to verify the validity of our technique when JPEG lossy compression is used. JPEG works well on some material like photographs, naturalistic artwork, but not so well on certain other material like lettering, simple cartoons, or line drawings [6].

Method: We compressed the 150 images in our test set to 50%, 30% and 10% and applied our technique to compare the original image with the compressed image. As in the previous experiment, the users were required to rank the images according to the quality they perceived. Some images were indistinguishable from the original even at 10% while some were distorted beyond 50%. Table 3 lists ten images, their *col_diff* values, and the user ranks for the 10%

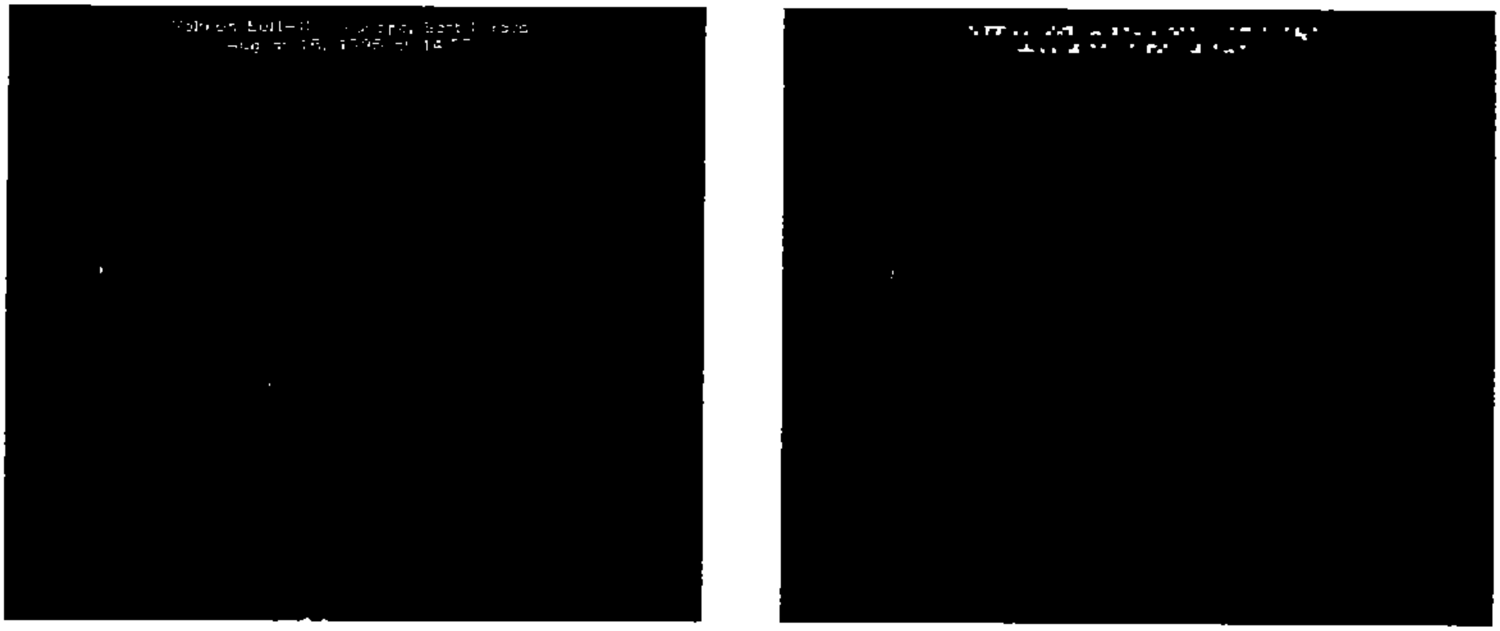

Figure 3: Original and Half Resolution Versions of the Red Image

JPEG compression. For convenience, the *col_diff* values are scaled by 10^7.

Discussion

As observed in Experiment 1, the user perceived quality correlates with the calculated *col_diff* values. Asteroid and Lion are the best images. Mars and Scenery become bad at 10% because of the "cube" or "checkerboard" effect introduced by the sequence of patches of colors. Red has a very high *col_diff* value at 10% due to the fact that the small dots of color contained in the image get highly distorted by the "cube" effect. Two other interesting observations can be made from the table. In two images, Galaxy and Panda, the *col_diff* value for 30% JPEG is *less* than that of 50% JPEG. This is counter-intuitive since 30% JPEG has more compression than the 50% JPEG image and one would expect *col_diff* to be less for the 50% JPEG image. For both images, both 30% and 50% JPEG versions are indistinguishable from the original to the human eye. 30% JPEG images should be used in both cases and will result in a significant reduction in communication time. Figures 4 and 5 show the original and half resolution versions for Lion and Mars images respectively. It can be observed that the 10% Lion image is indistinguishable from the original while the 10% Mars image is distinguishable.

4.3 Comparison with the Pixel-to-Pixel Matching Technique

Traditionally loss is measured by comparing the values of each individual pair of pixels [9]. Let X be an image and L be its loss-induced version. The loss in the root mean square (RMS) method is evaluated as follows:

$$RMS = \sqrt{\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}(X_{i,j} - L_{i,j})^2}$$

Name	50%	30%	10%	User Rank (for 10%)
Flowers	2.28	2.79	4.63	1
Curve	0.542	1.12	2.04	1
Asteroid	1.00	1.33	2.36	1
Panda	2.56	1.94	3.33	1
Flower	2.78	3.55	5.40	2
Lion	5.55	6.21	8.23	2
Scenery	2.18	3.69	12.19	5
Mars	6.14	8.63	17.21	6
Galaxy	65.32	16.26	87.85	7
Red	2.83	64.29	733.19	8

Table 3: *Col_diff* Values for Lossy Compressed Images Using JPEG to 50%, 30% and 10%

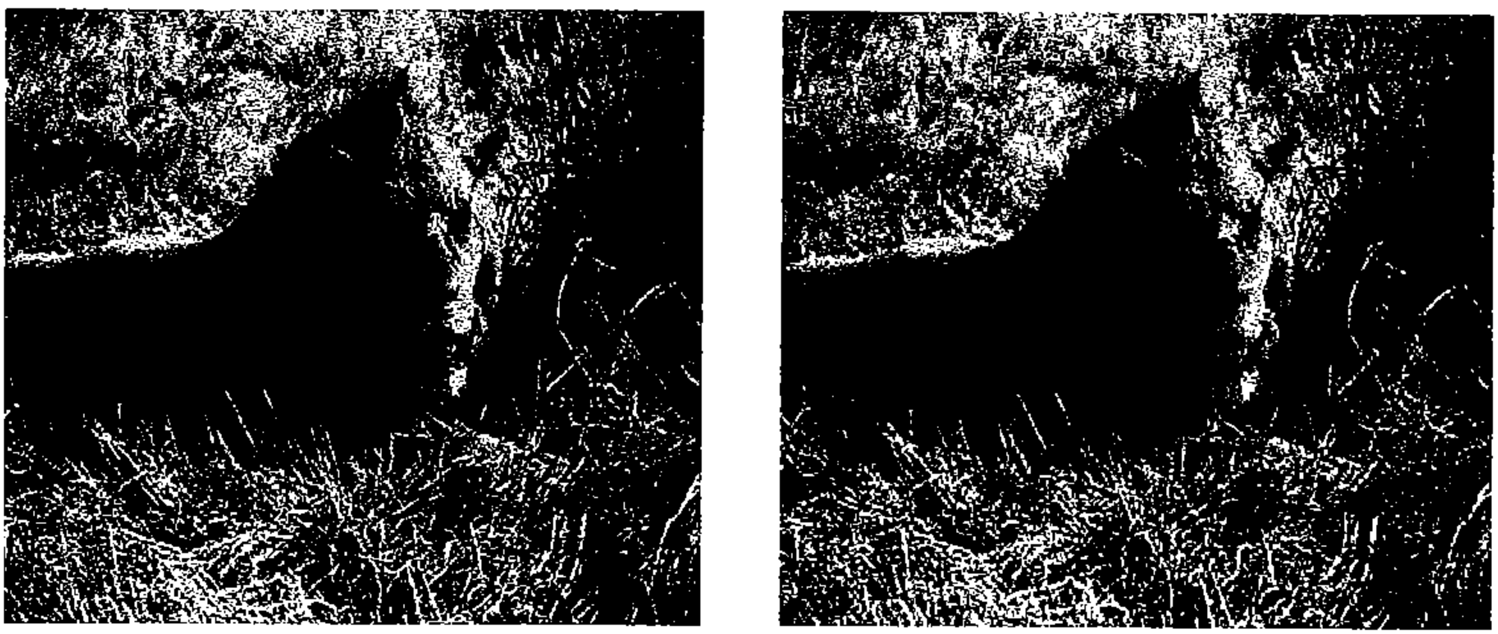

Figure 4: Original and 10% JPEG Versions of the Lion Image

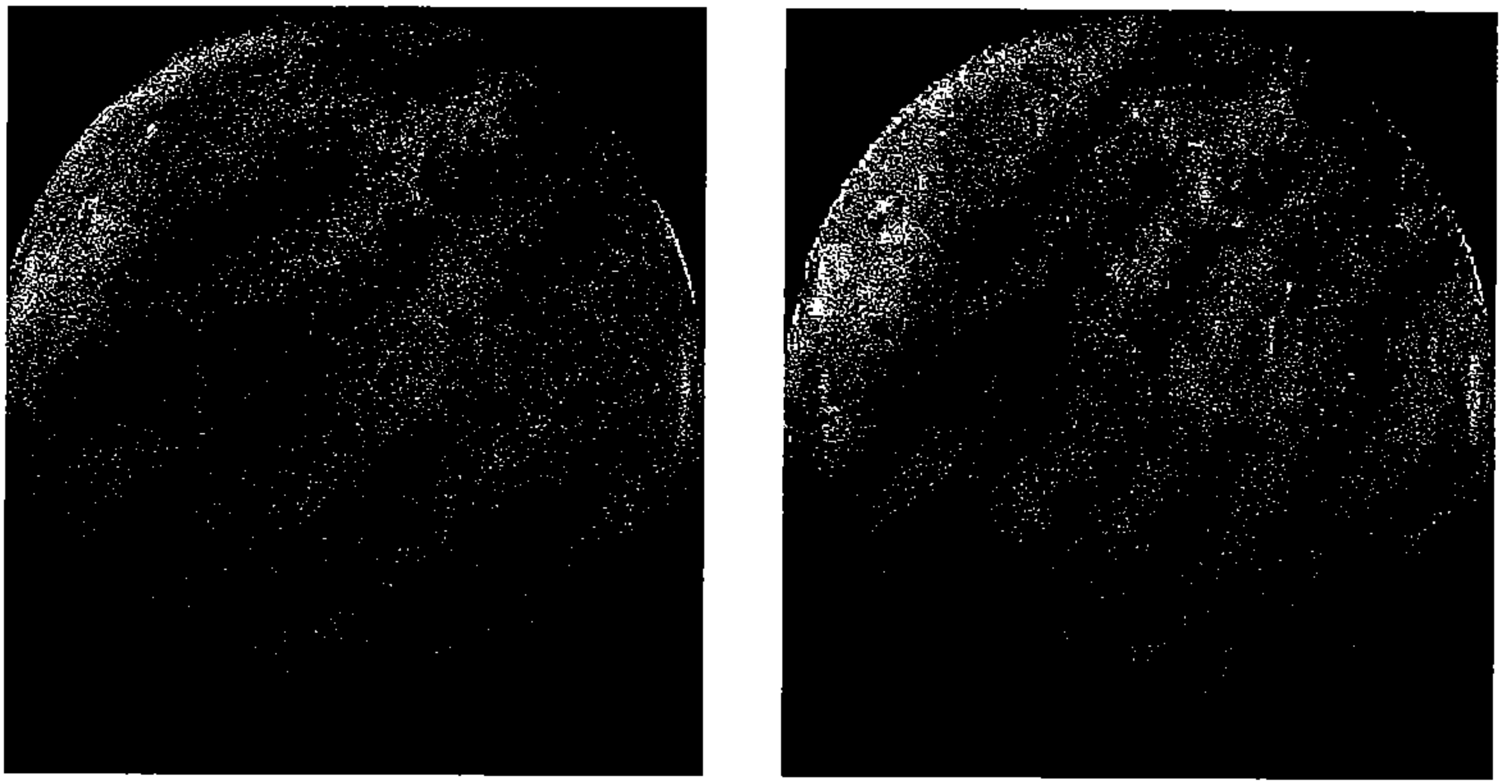

Figure 5: Original and 10% JPEG Versions of the Mars Image

This method is however, not an accurate reflection of the quality of the image with respect to the human eye. Table 4 presents the RMS values and the *col_diff* values calculated using our technique along with the user ranks. It can be clearly observed that the values produced by our technique correlate more closely with the user observed rank than the RMS values.

Name	RMS Value	*Col_diff* Value	User Rank
Curve	0.057982	2.04	1
Flowers	0.041668	4.63	1
Asteroid	0.046978	2.36	1
Panda	0.054082	3.33	1
Flower	0.029528	5.40	2
Lion	0.083567	8.23	2
Scenery	0.033784	12.19	5
Mars	0.042541	17.21	6
Galaxy	0.023562	87.85	7
Red	0.024711	733.19	8

Table 4: Comparison of RMS Values with *Col_diff* Values

According to the user evaluation, lossy compression of Lion and Asteroid result in very good images which are indistinguishable from the original. This is reflected in the *Col_diff* values but not in the RMS values. Lion has the highest value (among the ten images) of 0.084 and this does not correlate with our observations. Scenery results in bad quality image after compression but its RMS value is low. The RMS method is localized and fails to capture the differences from the image as a whole. Also, since there is no quantization step involved, differences not observable by the human eye are used to evaluate the RMS value.

5 Conclusions and Future Work

We have developed a technique based on color to quantify the data loss when a loss-inducing technique is applied to an image. Our focus is to estimate the quality of an image with respect to the human eye. Color is a natural parameter to be measured for that purpose since the human eye reacts differently to different combinations of colors. Our results indicate that changes in color provide a good measure of the data loss in the image particularly with respect to the human eye. Our results correlate with the user observations of the quality of images. Application of our technique will result in communicating only as much data as is required and not the 100% quality image which might be redundant in many cases. This will lead to better and more efficient usage of network resources.

Our future work involves refining this color-based technique further to quantify the loss of data as accurately as possible with respect to the loss as perceived by the human eye. We plan to model the stimulus each color has on the human eye and incorporate appropriate weights in our technique. This will lead to a more accurate estimate of what is a "good" image.

6 Acknowledgements

We would like to thank Dana Marie Farr for her assistance during the testing phase of this work.

References

[1] M. Annamalai and B. Bhargava. An evaluation of transmitting compressed images in a wide area network. Technical Report CSD 95-064, Department of Computer Sciences, Purdue University, October 1995.

[2] F. Arman, A. Hsu, and M. Chiu. Feature management for large video databases. In *Proceedings of Storage and Retrieval for Image and Video Databases*, pages 2–12, San Jose, CA, USA, February 1993. SPIE - The International Society for Optical Engineering.

[3] B. Bhargava and M. Annamalai. Communication costs in digital library databases. In *Lecture Notes in Computer Science Series (LNCS) 978, Database and Expert Systems Applications (DEXA '95)*, pages 1–13. Springer-Verlag, September 1995.

[4] Bharat Bhargava, Shunge Li, Shalab Goel, Chunying Xie, and Changsheng Xu. Performance studies for an adaptive video-conferencing system. In *Proceedings of the International Conference on Multimedia Information Systems (MULTIMEDIA 96), New Delhi, India.* IETE, February 1996.

[5] Y. Gong, H. Zhang, H. C. Chuan, and M. Sakauchi. An image database system with content capturing and fast image indexing abilities. In *Proceedings of the International Conference on Multimedia Computing and Systems; Boston, MA, USA*, pages 121–130, Boston, MA, USA, May 1994. IEEE Computer Society Press.

[6] JPEG-FAQ. at rtfm.mit.edu. From news.answers archive.

[7] D. C. Kay and J. R. Levine. *Graphics File Formats, second edition.* Windcrest/McGraw-Hill, 1995.

[8] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin. The QBIC project: Querying image by content using color, texture, and shape. In *Proceedings of Storage and Retrieval for Image and Video Databases*, pages 173–182, San Jose, CA, USA, February 1993. SPIE - The International Society for Optical Engineering.

[9] A. Rosenfeld and A. C. Kak. *Digital Picture Processing.* Academic Press, New York, 2nd Edition, 1982.

[10] M. J. Swain and D. H. Ballard. Color indexing. *Internationational Journal of Computer Vision*, 7(1):11–32, October 1991.

[11] G. K. Wallace. The JPEG still compression standard. *Communications of ACM*, 34(4):31–44, April 1991.