TIME-CONSTRAINED JPEG IMAGE TRANSMISSION VIA VARIABLE-QUALITY CODING*

Feng Shang and Albert Mo Kim Cheng
Real-Time Systems Laboratory
Department of Computer Science
The University of Houston-University Park
Houston, TX 77204-3010, U.S.A.

ABSTRACT

Real-time transmission of high-quality JPEG images is constrained by time, bandwidth, and transmission power in portable devices. To alleviate this problem, we designed, implemented, and evaluated a strategy that can adapt to different compression and transmission rates. (1) It gives important parts of an image higher priorities over unimportant parts. Therefore, the high-priority parts can achieve high image quality, while low-priority parts, with a slight sacrifice of quality, can achieve huge compression rate and thus save the power/energy of a low-power wireless system. (2) We also introduce the priority-driven scheduling approach into our coding algorithm, which makes the transmission of important parts earlier than other parts. Through a balanced trade-off between the available time and the image quality, this adaptive strategy can satisfy users with desired images quality and lead to a significant reduction of the important parts' deadline misses.

1. INTRODUCTION

Real-time transmission of high-quality images over low-bandwidth channels requires efficient image compression techniques. The most popular standard is the *Joint Photographic Experts Group* (JPEG). JPEG achieves compression ratios of 15 to. With a slight sacrifice of quality, 40-to-1 of compression, or more, is possible [5]. Although these standards provide a good compression performance, some problems still plague image transmission in the low-bitrate applications, especially wireless image/video communications. Two of these problems are introduced in the following.

The definition of the first international standard for low bit-rate, the new H263 standard, is below 64 kb/s [8]. However, this is still not suitable for wireless, low-bitrate networks due to the nature of wireless channels, which are more limited in terms of bandwidth and bit error rates when compared to the public telephone network. Therefore, new, robust and highly efficient coding algorithms will be necessary. Generally, an image/video coding scheme in the range of 8 to 12 kbps is desired [14] in the real world of wireless channels.

Obviously, there is a conflict between Quality and Compression for JPEG. However, we found that in most applications, users only care about some special objects in the image, while for others they just want to get a sweeping view. A good example is the image with a plane flying in the sky. Users are usually interested in the details of plane.

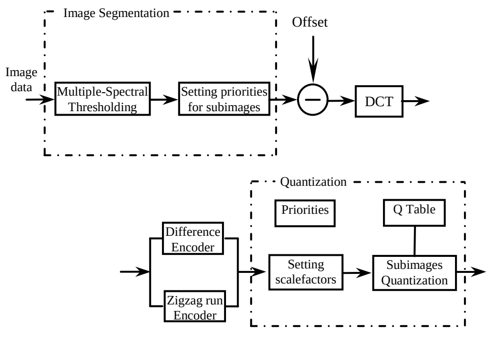
A lot of previous works deal with an image (or frame of video) as a whole with the same quality, which result in the image being either highly compressed (very coarse) or of high quality, but one cannot achieve both. In the JPEG format, at most four quantization tables, which control the quality of image, are occupied by the components Y, Cb, Cr. This makes it difficult to obtain an image with different quality in it. Our priority-driven coding approach applies quality on data unit, and reasonably discards DCT coefficients. It gives important parts of an image high priorities over unimportant parts.

We will apply priority-driven scheduling strategy [3] with Progressive JPEG standard for data transmission, which provides users with several levels of guaranteed *quality of service (QoS).* This provides a guarantee to users that the data transmitted is the data that they mostly desire to first receive regardless the power of the level.

This paper is organized as follows. Section 2 contains a diagram, which give you an overview of our strategy. This is followed by the details of the various implementation issues for the proposed strategy in Section 3, 4, 5. We include results for our strategy and show an improved performance in Section 6. We finally conclude in Section 7.

2. OVERVIEW OF PRIORITY-DRIVEN CODING OF PROGRESSIVE JPEG

Figure 1 gives a diagram of our strategy based on the JPEG encoder. We will discuss the strategy step by step in the following subsections.



* This material is supported in part by the National Science Foundation under Award No. IRI-9526004 and a grant from the Institute of Space Systems Operations.

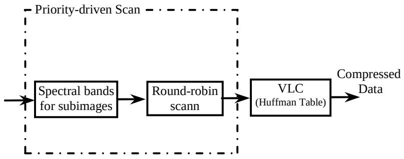


Figure 1: Diagram of the priority-driven coding strategy.

3. MULTIPLE-SPECTRAL THRESHOLDING

For the part of image processing in our strategy, we employ the multiple-spectral thresholding method to divide an image into several subimages. The multiple-spectral thresholding method can easily be done in real time using specialized hardware. Figure 2 gives a result after segmentation for one part of whole image.

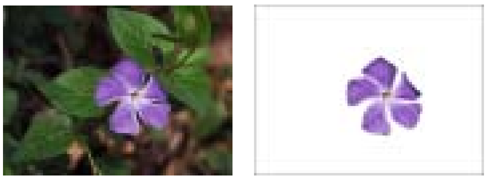


Figure 2: Original image and one part of the image after segmentation.

4. COMPRESSION WITH DIFFERENT QUALITIES

The fact that the JPEG standard only provides a limited number of quantization tables for one image, usually one table for one component, results in an image uniformly with the same quality. Now the question is how to make the image with different qualities for different subimages? The answer is, instead of scaling the quantization table, we keep it unchanged, while scaling every data unit after DCT according to their quality when we encode the image. The following equation is used to quantize the data unit.

$$QuantizedValue = Round\left(\frac{Coefficient}{Round(QuantumValue \times scalefactor)} \times scalefactor\right) \qquad (1)$$

Here, Quantum Value is originally from the quantization table and we will not scale it like convention. Quantized Value is the final data for every data unit and it is obtained after two times' nearest integer truncation. This will lead to the same results with the approach of scaling quantization table after decoding, since both of them have twice the nearest integer truncation. This difference is that the original one applies one on data units, one on quantization table, but ours applies both on data units.

We give an example here. We use the data unit after DCT from an image in Figure 3. Suppose we assign this data unit with quality 40. We now show the two methods at following.

$$\begin{bmatrix} -603 & 203 & 11 & 45 & -30 & -14 & -14 & -7 \\ -108 & -93 & 10 & 49 & 27 & 6 & 8 & 2 \\ -42 & -20 & -6 & 16 & 17 & 9 & 3 & 3 \\ 56 & 69 & 7 & -25 & -10 & -5 & -2 & -2 \\ -33 & -21 & 17 & 8 & 3 & -4 & -5 & -3 \\ -16 & -14 & 8 & 2 & -4 & -2 & 1 & 1 \\ 0 & -5 & -6 & -1 & 2 & 3 & 1 & 1 \\ 8 & 5 & -6 & -9 & 0 & 3 & 3 & 2 \end{bmatrix}$$

Figure 3: data unit after DCT.

(1) Conventional Approach

- *Encoding*

We first compute the *scale factor* = $2^{1.2}$ = *2.3*. After scaling the original quantization table, we obtain the quantization table, and the data unit after quantization which shown in Figure 4.

$$\begin{bmatrix} 37 & 25 & 23 & 37 & 55 & 92 & 117 & 140 \\ 28 & 28 & 32 & 43 & 60 & 133 & 138 & 126 \\ 32 & 30 & 37 & 55 & 92 & 131 & 158 & 129 \\ 32 & 39 & 51 & 67 & 117 & 200 & 184 & 143 \\ 41 & 51 & 85 & 129 & 156 & 251 & 237 & 177 \\ 55 & 80 & 126 & 147 & 186 & 239 & 260 & 211 \\ 113 & 147 & 179 & 200 & 237 & 278 & 276 & 232 \\ 166 & 212 & 218 & 225 & 258 & 230 & 237 & 228 \end{bmatrix} \quad \begin{bmatrix} -16 & 8 & 0 & 1 & 0 & 0 & 0 & 0 \\ -4 & -3 & 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 4: table after scaling and data unit after quantization.

- *Decoding*

When decoding, we will use data unit to multiply the quantization table in Figure 4, then we obtain the data unit after DCT shown in Figure 5:

$$\begin{bmatrix} -592 & 200 & 0 & 23 & 0 & 0 & 0 & 0 \\ -112 & -84 & 0 & 43 & 0 & 0 & 0 & 0 \\ -32 & -30 & 0 & 0 & 0 & 0 & 0 & 0 \\ 64 & 78 & 0 & 0 & 0 & 0 & 0 & 0 \\ -41 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 5: Data unit after DCT.

(2) OUR APPROACH

- *Encodeing*

With our approach, through equation (1), we directly obtain the data unit after quantization shown in Figure 6. The quantization table remains unchanged.

$$\begin{bmatrix} -37 & 18 & 0 & 2 & 0 & 0 & 0 & 0 \\ -9 & -7 & 0 & 2 & 0 & 0 & 0 & 0 \\ -2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 5 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 6: Data unit after quantization.

- *Decodeing*

When decoding, we use the data in Figure 6 to multiply the unchanged quantization table, then we obtain the data unit after DCT shown in Figure 7:

$$\begin{bmatrix} -592 & 198 & 0 & 32 & 0 & 0 & 0 & 0 \\ -108 & -84 & 0 & 38 & 0 & 0 & 0 & 0 \\ -28 & -26 & 0 & 0 & 0 & 0 & 0 & 0 \\ 70 & 105 & 0 & 0 & 0 & 0 & 0 & 0 \\ -36 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 7: Data unit after DCT.

We have found that the data in Figure 5 and Figure 7 have the same approximation to the original data in Figure 3.

5. PRIORITY-DRIVEN SCHEDULING STRATEGY

The progressive JPEG refers to the encoding of the DCT coefficients in multiple scans. It gives a good structure to apply our priority-driven scheduling approach. It makes the data units with high priority value send more data in the earlier scans than data units with low priority. For example, we can assign the spectral ranges of data units with high priority value

0-3 3-8 8-15 15-30 30-63

as the spectral bands for the scan, comparing the spectral ranges of data units with low priority value

0-1 1-5 5-10 10-20 20-30 30-63

as the spectral bands for the scan.

We make every scan a round robin, and give each data unit a fraction of the data in one round robin. The fraction of each data unit is allocated according to its priority value. The data units with high priority value have been allocated more fractions, which means that it will send more data in earlier scans than data units with low priority. The user determines how many data items are encoded in every scan, but this is mainly determined by the bandwidth of the network through which the image is transmitted. With a high bandwidth, every scan can encode more data. We can make the amount of data encoded in one scan equal to or less than the bandwidth of the network. The following example illustrates three data units with round-robin scheduling. We consider coding these three data units coding as three sets of jobs,

$J_1 = \{ J_{1,1}, J_{1,2}, J_{1,3}, \dots \};$

$J_2 = \{ J_{2,1}, J_{2,2}, J_{2,3}, \dots \};$

$J_3 = \{ J_{3,1}, J_{3,2}, J_{3,3}, \dots \}.$

Suppose that $J_{1,1}$ is the predecessor of $J_{1,2}$, and $J_{2,1}$ is the predecessor of $J_{2,2}$.

Assume that J_1 has highest priority among them and J_3 has the lowest priority, then we assign the spectral ranges of three data units according to their priority value as

J_1: 0-4 5-9 10-63

J_2: 0-2 3-7 8-15 16-63

J_2: 0-1 2-3 4-11 12-30 30-63.

We find that these spectral ranges are just mapping with the task sets of J_1, J_2, and J_3, finally we get the following job lists:

$J_1 = \{ J_{1,1}(0\text{-}4), J_{1,2}(5\text{-}9), J_{1,3}(10\text{-}63) \};$

$J_2 = \{ J_{2,1}(0\text{-}2), J_{2,2}(3\text{-}7), J_{2,3}(8\text{-}15), J_{2,4}(16\text{-}63) \};$

$J_3 = \{ J_{3,1}(0\text{-}1), J_{3,2}(2\text{-}3), J_{3,3}(4\text{-}11), J_{3,4}(12\text{-}30), J_{3,5}(30\text{-}63) \};$

6. EXPERIMENTAL RESULTS

We tested several JPEG files of different kind of sizes using our strategy in order to compare it with the existing algorithm and some representative results are presented in this section. Subsection 6.1 presents the results obtained for the part of strategy that deals with compression with different qualities for the encoder output, and shows the high compression ratio. Subsection 6.2 presents the results for the part dealing with priority-driven transmission strategy.

6.1 Compression Ratio with Different Qualities

After sub-dividing the image with size of 60 kbytes into three regions, we code it with three different qualities as shown in Figure 8.

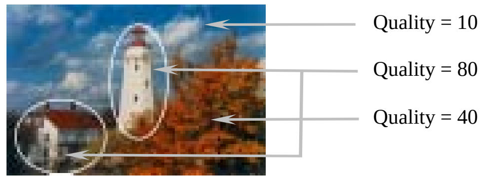


Figure 8: The qualities assigned to regions of the image.

Figure 9 shows the resulting image with assigned qualities. After compression, the file size becomes 12 kbytes. The compression ratio is 5.0.

Figure 9: The image with three regions of different qualities.

The following table summarizes the performance of our approach for three representative images.

	Original (Kbytes)	Final(Kbytes)	Compression Ratio
Image 1	1772	198	9.0
Image 2	365	63	5.8
Image 3	60	12	5.0

Table 1: The compression ratio result.

The size of the JPEG files that we generally use is around 100 kbytes. With our strategy, we obtain a file with a size of less than 20 kbytes, thus a compression ratio of about 5 can be achieved, a good result.

6.2 Phases of Priority-driven Coding

Table 2 shows the jobs we have assigned to the previous image. We assume that we transmit the data in four rounds, then we will have four phases for displaying images with different qualities.

Round / ***Job***	**1**	**2**	**3**	**4**
1 (q=80)	$J_{1,1}(0\text{-}0)$	$J_{1,2}(1\text{-}3)$	$J_{1,3}(4\text{-}10)$	$J_{1,4}(11\text{-}63)$
2 (q=40)	×	$J_{2,1}(0\text{-}0)$	$J_{2,2}(1\text{-}4)$	$J_{2,3}(5\text{-}63)$
3 (q=10)	×	×	$J_{3,1}(0\text{-}0)$	$J_{3,2}(1\text{-}63)$

Table 2: Priority-driven jobs.

Figure 10 shows the first phase of displaying image. From the above table, we know that only part of the data of the region with quality 80 has been transmitted.

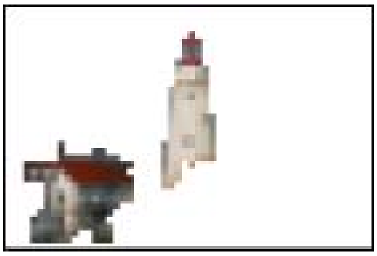


Figure 10: Image displayed after the first round.

Figure 11 shows the second phase image we displayed. This time two regions' data have been transmitted, and the region with quality 80 is refined.

Figure 11: Image displayed after the second round.

Figure 12 shows the third phase image we displayed. This time all regions' data have been transmitted, and the data of the regions already transmitted were refined.

Figure 12: Image displayed after third round.

After the final round, all regions' data have been transmitted, and the data of all regions were refined (See Figure 9).

7. CONCLUSION

Transmitting JPEG images under time, power, and bandwidth constraints requires deleting certain image details from the original image prior to its transmission. Our adaptive strategy assigns important subimages of an image higher priority over unimportant subimages. This leads to a higher compression rate, so fewer bits need to be transmitted for a given image, and thus saving the power/energy of low-power wireless systems. Based on the JPEG standard, our strategy further achieves compression ratios of about 5. Therefore, the combined compression ratios will be about 150. In general, after image compression with our strategy, the size of the images commonly used will be equal to or less than 20 kbyes. This is more suitable for wireless transmission when the bandwidth is limited. We also introduce the priority-driven scheduling approach into our image coding strategy. This strategy schedules the important subimages for transmission earlier with more data than the other less important parts. When it is impossible to transmit the entire image on time, our strategy will discard some data of unimportant subimages. Through a balance trade-off between the available time and the image quality when the available time is not sufficient for transmission of the entire image due to low available power/energy, our adaptive strategy can satisfy users with desired images quality and lead to a significant reduction of deadline misses of the important data, thus meeting a desired quality of service.

8. REFERENCES

[1] Joint Photographic Experts Group, ISO/IEC/JTC1/SC2 /WG8, "JPEG Technical Specification, Revision 8", Aug. 1990.

[2] John Miano, "Compressed Image File Formats, Addison-Wesley", Reading, Massachusetts, 1999.

[3] Jane W. S. Liu, "Real-Time Systems", Prentice-Hall, Inc., 2000.

[4] X. Chen and A. M. K. Cheng, "An Imprecise Algorithm for Real-Time Compressed Image/Video Transmission," *Proc. 6th Intl. Conf. on Computer Communications and Networks*, Las Vegas, Nevada, Sept. 1997.

[5] Milan Sonka, Vaclav Hlavac and Roger Boyle, "Image Processing, Analysis and Machine Vision", 2nd Edition, Brooks/Cole Publishing Company, 1999.

[6] Ziv, J. and Lempel, A., "Compression of Individual Sequences via Variable-Rate Coding," *IEE Transactions on Information Theory*, Volume 24, Number 5, September 1978, pages 530-536.

[7] B. G. Haskell, A. Puri, and A. N. Netravali, "Digital Video: An Introduction to MPEG-2," Mew York: Chapman and Hall, 1997.

[8] Y. Wang, Q. Zhu, and L. Shaw, "Maximally smooth image recovery in transform coding," *IEEE Transaction Communication*, vol. 41, October 1993.

[9] J. Albanese, A. amd Blomer, J. Edmonds, M. Luby, and M. Sudan, "Priority encoded transmission," *IEEE Transaction On Information Theory*. November 1996.

[10] X. Huang and A. Cheng, "Applying Imprecise Computation Algorithms to Real-Time Image and Video Transmission," *Proc. IEEE-CS RTTS*, 1995.

[11] L. D. Nguyen and A. M. K. Cheng, "An Imprecise Real-Time Image Magnification Algorithm," *Proceedings of International Symposium on Multimedia Systems*, Mar. 1996.

[12] V. K. Goyal, J. Kovacevic, R. Arean, and M. Vetterli, "Multiple description transform coding of images," *Proceedings of IEEE ICIP*, October, Chicago, IL, 1998.

[13] Daniel Grove Sachs, Raghqvan Anand and Lanuan Ramchandran, "Wireless Image Transmission Using Multiple-Description Based Concatenated Codes," *IEEE*, 2000.

[14] C. Wong and A. M. K. Cheng, "An Approach for Imprecise Transmission of TIFF Image Files Through Congested Real-Time ATM Networks," *Proc. 22nd Intl. Conf. on Local Computer Networks*, Minneapolis, MN, Nov. 1997.