Title	Compound document compression with model-based biased reconstruction
Author(s)	Lam, EY
Citation	Journal Of Electronic Imaging, 2004, v. 13 n. 1, p. 191-197
Issued Date	2004
URL	http://hdl.handle.net/10722/42949
Rights	Creative Commons: Attribution 3.0 Hong Kong License

Compound document compression with model-based biased reconstruction

Edmund Y. Lam
University of Hong Kong
Department of Electrical and Electronic Engineering
Hong Kong
E-mail: elam@eee.hku.hk

Abstract. *The usefulness of electronic document delivery and archives rests in large part on advances in compression technology. Documents can contain complex layouts with different data types, such as text and images, having different statistical characteristics. To achieve better image quality, it is important to make use of such characteristics in compression. We exploit the transform coefficient distributions for text and images. We show that the scheme in baseline JPEG does not lead to minimum mean-square error if we have models of these coefficients. Instead, we discuss an algorithm designed for this performance that involves first classifying the blocks, and then estimating the parameters to enable a biased reconstruction in the decompression value. Simulation results are shown to validate the advantages of this method.* © 2004 SPIE and IS&T.
[DOI: 10.1117/1.1631317]

1 Introduction

With the rapid increase in signal bandwidth and memory capacity, electronic document delivery has been gaining popularity in recent years. In an office environment, for example, many documents are simply scanned and then sent electronically to a list of recipients rather than photocopied and distributed to their desks. It is therefore very important to be able to compress them efficiently.

Many of these documents contain a mixture of data types, such as natural images, text, line art, and background. It is known that these data types have different statistical characteristics. Specialized compression algorithms have been developed for text (such as JBIG and JBIG2[1]) and images (such as JPEG[2] and JPEG 2000[3]), respectively. Some compression standards, such as DjVu,[4] have provisions for using multiple compression schemes for text and image compression. However, sometimes we only use a single compression method for the entire image, where JPEG is the most common. This is especially true of low-end hardware and software that need to reduce the codec complexity. Fortunately, in JPEG we still have some control on the parameters of the algorithm to adapt it for different image types. Compressing compound documents by varying these parameters has received significant attention in recent years.[5–7]

As a brief review, the JPEG algorithm is summarized in Fig. 1. The image is first divided into 8×8 nonoverlapping blocks, and each block is subjected to a discrete cosine transform (DCT). The coefficients are then quantized according to the quantization matrix Q_e, by rounding off the quotients when they are divided entrywise by Q_e. The quantized coefficients are then entropy coded before transmission. The decoder reverses the process for Huffman coding, dequantizes the coefficients by multiplying entrywise with the matrix Q_d, and computes the inverse DCT. The compression is lossy because of the quantization process.

It is common to use the same quantization matrix for both encoding and decoding, i.e., $Q_e = Q_d$. The JPEG committee suggests the matrix, as shown in Fig. 2 for both Q_e and Q_d, take into account some of the human visual system properties, although its use is strictly voluntary. However, setting the quantization and dequantization matrices to be the same does not give the best image quality, if we have information about the statistics of the transform coefficients.

Let the DCT coefficients in a block be $I(u,v)$, where u and v are the spatial frequencies in the horizontal and vertical directions, both ranging from 0 to 7. Using $I_q(u,v)$ to denote the quantized coefficients and $I_n(u,v)$ for the quantization noise, we see that they are related by

$$\frac{I(u,v)}{Q_e(u,v)} = I_q(u,v) + I_n(u,v). \qquad (1)$$

From a statistical point of view, $I_n(u,v)$ is a random variable with $-0.5 \leqslant I_n(u,v) < 0.5$. Since we only transmit $I_q(u,v)$, for decoding, we have

$$\hat{I}(u,v) = Q_d(u,v) I_q(u,v). \qquad (2)$$

To compare the original and decompressed images, we normally would have to calculate the mean-square error (MSE) in the space domain. However, because of the unitary nature of the 2-D DCT, we could perform the calculation in the DCT domain because of Parseval's theorem.[8] Therefore,

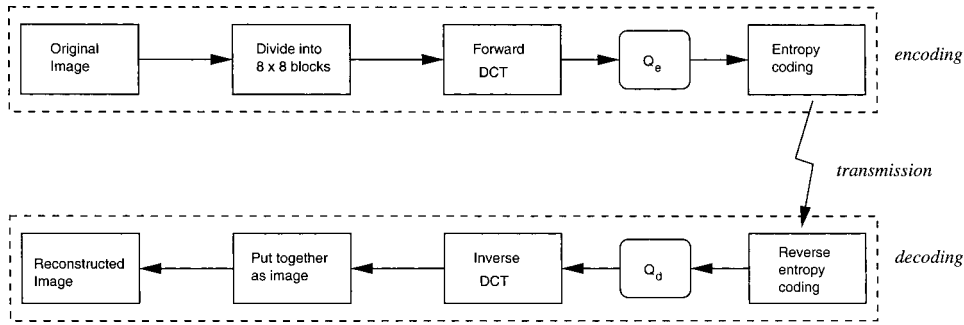**Fig. 1** Block diagram of a block-based linear transform for image compression.

$$\text{MSE} = \sum_{u=0}^{7} \sum_{v=0}^{7} [I(u,v) - \hat{I}(u,v)]^2$$

$$= \sum_{u=0}^{7} \sum_{v=0}^{7} [(Q_e - Q_d)I_q + Q_e I_n]^2, \qquad (3)$$

where it is understood that Q_e, Q_d, I_q, and I_n all have arguments (u,v). From this equation, we can see that setting $Q_e = Q_d$ does not necessarily lead to the minimum MSE. This has been exploited in Ref. 9 to enhance image quality.

We employ a different methodology. Suppose we bias the reconstruction by B, i.e., the decompression is computed using

$$\tilde{I}(u,v) = Q_d(u,v)I_q(u,v) - B(u,v). \qquad (4)$$

Therefore,

$$\text{MSE} = \sum_{u=0}^{7} \sum_{v=0}^{7} [(Q_e - Q_d)I_q + (Q_e I_n - B)]^2$$

$$= \sum_{u=0}^{7} \sum_{v=0}^{7} (Q_e I_n - B)^2, \qquad (5)$$

if we set $Q_e = Q_d$. This is minimum when

```
16 11 10 16  24  40  51  61

12 12 14 19  26  58  60  55

14 13 16 24  40  57  69  56

14 17 22 29  51  87  80  62

18 22 37 56  68 109 103  77

24 35 55 64  81 104 113  92

49 64 78 87 103 121 120 101

72 92 95 98 112 100 103  99
```

Fig. 2 The recommended JPEG quantization matrix.

$$B(u,v) = Q_e(u,v)E[I_n(u,v)], \qquad (6)$$

where $E[\cdot]$ denotes expectation. In essence, $B(u,v)$ moves the decoding value $\tilde{I}(u,v)$ to be the centroid of the code block.[10] In the absence of any known distribution of $I_n(u,v)$, we may assume it is uniform between ± 0.5, and therefore $B(u,v) = 0$. However, in the next section we construct an image model that shows that $I_n(u,v)$ behaves differently for text and images. We can then make use of this knowledge to enhance compound document compression.

2 Doubly Stochastic Image Model

To compute the centroid in each code block, we first need to have a distribution model for the transform coefficients. Aside from many empirical studies that employ goodness-of-fit techniques for the distributions,[11–14] a doubly stochastic model has been shown to provide a solid mathematical foundation for this purpose.[15,16]

In this model, the distribution is computed in a two-step process. First, within each 8×8 block used for DCT, the pixels are deemed to be identically distributed. They do not need to be independent, as long as the correlation between adjacent pixels is not too big compared with the block size. Let $i(p,q)$ denote a pixel, $p = 0, \ldots 7$, $q = 0, \ldots 7$, within the block. The DCT is computed with the equation

$$I(u,v) = \frac{C(u)C(v)}{4} \sum_{p=0}^{7} \sum_{q=0}^{7} \left\{ i(p,q) \cos\left[\frac{(2p+1)\pi u}{16}\right] \right.$$

$$\left. \times \cos\left[\frac{(2q+1)\pi v}{16}\right] \right\}, \qquad (7)$$

with

$$C(v) = \begin{cases} \dfrac{1}{\sqrt{2}} & \text{for } v = 0 \\[2mm] 1 & \text{for } v > 0 \end{cases}. \qquad (8)$$

Equation (7) can be interpreted as a weighted sum of identically distributed random variables. By the law of large numbers, the DCT coefficient is approximately Gaussian distributed. Let σ^2 denote the variance. We therefore have

$$\mathcal{P}[I(u,v)|\sigma^2] = \frac{1}{\sqrt{2\pi}\sigma}\exp\left\{-\frac{I^2(u,v)}{2\sigma^2}\right\}. \qquad (9)$$

Second, across different blocks in the image, we consider the block variance to be a stochastic quantity itself. This distribution varies with different image types. For natural images, it resembles an exponential distribution.[15] For text, a uniform distribution is a better model.[16] In both cases, we can compute the DCT coefficient distribution using the equation

$$\mathcal{P}[I(u,v)] = \int_0^\infty \mathcal{P}[I(u,v)|\sigma^2]\mathcal{P}(\sigma^2)d(\sigma^2). \qquad (10)$$

If we put the exponential distribution and Eq. (9) into the previous equation, i.e., with

$$\mathcal{P}(\sigma^2) = \lambda\exp\{-\lambda(\sigma^2)\}, \qquad (11)$$

after some manipulation,[15] we have

$$\mathcal{P}[I(u,v)] = \frac{\sqrt{2\lambda}}{2}\exp\{-\sqrt{2\lambda}|I(u,v)|\}. \qquad (12)$$

Therefore, the distribution of the DCT coefficients for natural images is Laplacian. If we put the uniform distribution and Eq. (9) into Eq. (10), i.e., with

$$\mathcal{P}(\sigma^2) = \frac{1}{t-s}, \qquad (13)$$

for $s \leq \sigma^2 \leq t$, the result does not produce a closed-form solution. However, we can evaluate the integral numerically. The result is shown to resemble a Gaussian distribution.[16] In both cases, we can conclude that if $I(u,v)$ is not a uniform distribution, $I_n(u,v)$ would also be non-uniform with the quantization scheme in Eq. (1). A biased reconstruction from mid-point to the centroid of each code block will therefore enhance the decompression, with the details presented in the next section.

3 Biased Reconstruction

We can now compute the centroid of the code block. Without loss of generality, assume $I_q(u,v)$ is positive. Because the probability density functions of $I(u,v)$ are zero mean and symmetric, if $I_q(u,v)=0$ we have $B(u,v)=0$, i.e., no bias is necessary. The case for $I_q(u,v)<0$ is the mirror image of $I_q(u,v)>0$. The boundaries of the code block are
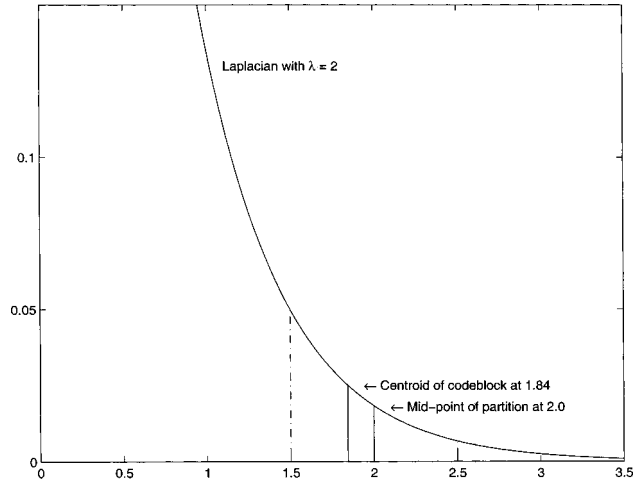

Fig. 3 Centroid and mid-point for a code block in a Laplacian distribution.

therefore $a=(I_q-0.5)Q_e$ and $b=(I_q+0.5)Q_e$, where we drop the arguments (u,v) when no ambiguity arises. For a Laplacian distribution, the centroid is

$$
\begin{aligned}
\hat{x} &= \frac{\displaystyle\int_a^b x\,\frac{\lambda}{2}\exp\{-\lambda x\}dx}{\displaystyle\int_a^b \frac{\lambda}{2}\exp\{-\lambda x\}dx} \\[2mm]
&= \frac{\dfrac{1}{2}\left[-x\exp\{-\lambda x\}-\dfrac{1}{\lambda}\exp\{-\lambda x\}\right]_a^b}{\frac{1}{2}\exp\{-\lambda a\}-\frac{1}{2}\exp\{-\lambda b\}} \\[2mm]
&= \frac{a\exp\{-\lambda a\}-b\exp\{-\lambda b\}}{\exp\{-\lambda a\}-\exp\{-\lambda b\}}+\frac{1}{\lambda} \\[2mm]
&= \frac{a+b}{2}+\frac{a-b}{2}\frac{\exp\{-\lambda a\}+\exp\{-\lambda b\}}{\exp\{-\lambda a\}-\exp\{-\lambda b\}}+\frac{1}{\lambda} \\[2mm]
&= I_q Q_e - \frac{Q_e}{2}\coth\left(\frac{\lambda Q_e}{2}\right)+\frac{1}{\lambda}.
\end{aligned}
\qquad (14)
$$

The bias term is therefore

$$B = \left[\frac{1}{2}\coth\left(\frac{\lambda Q_e}{2}\right)-\frac{1}{\lambda Q_e}\right]Q_e. \qquad (15)$$

Note that $B>0$ because $\coth(x)>1/x$ for positive x. An example is shown in Fig. 3.

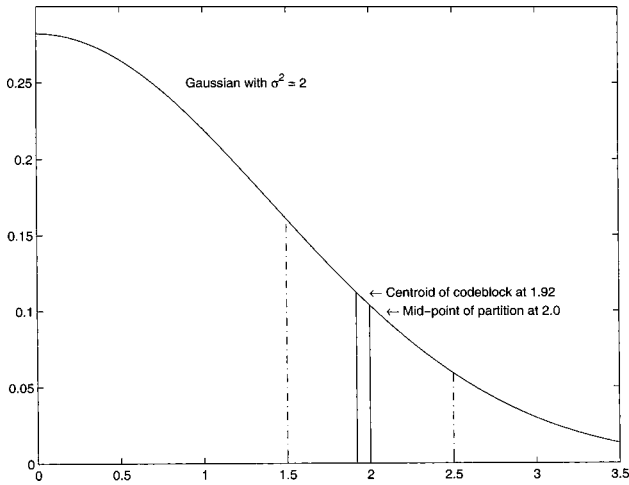On the other hand, for a Gaussian distribution, the centroid is

Fig. 4 Centroid and mid-point for a code block in a Gaussian distribution.

Fig. 5 An example of a mixed document.

$$\hat{x} = \frac{\int_a^b x \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{x^2}{2\sigma^2}\right\} dx}{\int_a^b \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{x^2}{2\sigma^2}\right\} dx}$$

$$= \frac{\frac{1}{2\sqrt{2\pi}\sigma} \int_{a^2}^{b^2} \exp\left\{-\frac{y}{2\sigma^2}\right\} dy}{Q\left(\frac{a}{\sigma}\right) - Q\left(\frac{b}{\sigma}\right)}$$

$$= \frac{\sigma\left(\exp\left\{-\frac{(I_q-\frac{1}{2})^2 Q_e^2}{2\sigma^2}\right\} - \exp\left\{-\frac{(I_q+\frac{1}{2})^2 Q_e^2}{2\sigma^2}\right\}\right)}{\sqrt{2\pi}\left[Q\left(\frac{(I_q-\frac{1}{2})Q_e}{\sigma}\right) - Q\left(\frac{(I_q+\frac{1}{2})Q_e}{\sigma}\right)\right]},$$

$$\tag{16}$$

where the function $Q(x)$ is the Q function defined as[17]

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty \exp\{-t^2\} dt. \tag{17}$$

Therefore, the bias term is

$$B = I_q Q_e$$
$$- \frac{\sigma\left(\exp\left\{-\frac{(I_q-\frac{1}{2})^2 Q_e^2}{2\sigma^2}\right\} - \exp\left\{-\frac{(I_q+\frac{1}{2})^2 Q_e^2}{2\sigma^2}\right\}\right)}{\sqrt{2\pi}\left[Q\left(\frac{(I_q-\frac{1}{2})Q_e}{\sigma}\right) - Q\left(\frac{(I_q+\frac{1}{2})Q_e}{\sigma}\right)\right]}.$$

$$\tag{18}$$

Note that this bias is a function of I_q, and is also always positive. An example is shown in Fig. 4.

4 Compound Document Compression

We are now ready to modify the JPEG algorithm to achieve optimal decompression for a compound document. To begin with, we need to distinguish between text and image regions in a compound document. Unlike many traditional image segmentation algorithms, we want the classification to be done on a block basis rather than at the pixel level. This has the added advantage of minimizing both memory and processing requirements.

Let $D(j)$ be a discriminant function on the j'th block, which indicates whether the block should be classified as text or as image. We simplify the method proposed in Ref. 5 so that each $D(j)$ is independent of its neighbors for ease of computation and a potential for parallelization. We compute $D(j)$ as a function of the 63 AC coefficients with

$$D(j) = \underbrace{\sum_{u=0}^{7} \sum_{v=0}^{7}}_{\text{except } u=v=0} g[I_q(u,v;j)], \tag{19}$$

where

$$g(x) = \begin{cases} \log_2(|x|) + 4 & \text{if } |x| \neq 0 \\ 0 & \text{otherwise} \end{cases}. \tag{20}$$

The constant 4 is an estimate of the average number of bits needed to encode a nonzero coefficient.[5] $I_q(u,v;j)$ denotes the quantized DCT coefficient at the (u,v) subband for block j. A higher value in $D(j)$ indicates that this block is more likely to be text. In fact, we use the value $D(j)$ to decide on the nature of the block j as follows:
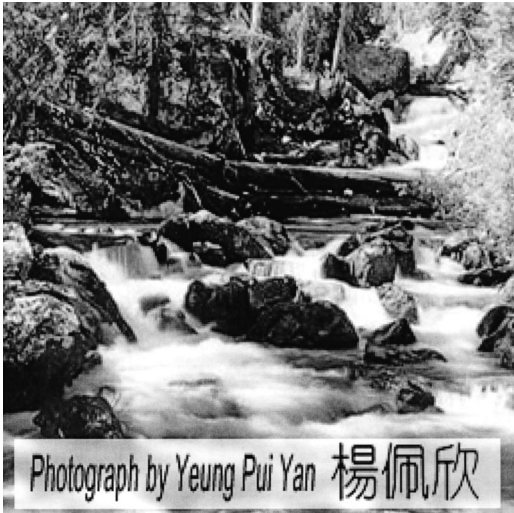
Fig. 6 An example of text inside an image.

$D(j) \approx 0 \Rightarrow \text{block is background}$

$D(j) < T_0 \Rightarrow \text{block is image.}$ (21)

$D(j) \geq T_0 \Rightarrow \text{block is text}$

T_0 is a threshold parameter. Essentially, if the block is background, there is no need for any adjustment to the DCT coefficients. When the block is classified as an image, we use a Laplacian probability density function to model the AC coefficients. When the block is classified as text, we use a Gaussian probability density function to model the AC coefficients. In both cases, we use the scheme described in the previous section to adjust the decompression value, using Eqs. (14) and (16). However, the issue remains as to how to estimate the parameters for either the Laplacian or Gaussian distributions.

For Laplacian, the maximum likelihood parameter estimation for λ given the observations $I(u,v;j)$ is[18]

$$\lambda(u,v)^{\text{ML}} = \frac{N_j}{\sum_{j=1}^{N_j} |I(u,v;j)|},$$ (22)

where N_j is the total number of blocks. This, however, requires access to the unquantized values. In the decoder, we only have the quantized values $I_q(u,v)$. We can put $I_q(u,v;j)$ into Eq. (22) to obtain a rough estimate of $\lambda(u,v)$. Since more numbers are quantized toward 0, this

Table 1 Simulation results for various coefficient models.

	SNR	
	Fig. 5	Fig. 6
Normal dequantization	20.40 dB	17.75 dB
Laplacian model	20.47 dB	18.05 dB
Gaussian model	20.73 dB	17.93 dB
Mixed model	20.74 dB	18.05 dB

calculation will generally result in an overestimation of $\lambda(u,v)$, and we can reduce it by a constant factor in practice.

As for Gaussian, the maximum likelihood parameter estimation is[18]

$$\sigma(u,v)^{\text{ML}} = \left\{ \sum_{j=1}^{N_j} \frac{I(u,v;j)^2}{N_j} - \left[\sum_{j=1}^{N_j} \frac{I(u,v;j)}{N_j} \right]^2 \right\}^{1/2}.$$ (23)

We can put $I_q(u,v;j)$ into Eq. (23) to obtain a rough estimate of $\sigma(u,v)$, and then increase it by a constant factor.

5 Simulation

To test the ideas proposed in this work, we evaluate the performance of the algorithm on a couple of test images. Figure 5 shows one such image, with predominantly text and an embedded image. The figure is of size 512×512 pixels. We also tested with documents such as Fig. 6, which is mostly image but with embedded text. This latter one is 256×256 pixels.

We tested the images in a couple of ways. First, we use the decompression mechanism in the baseline JPEG, which does not assume any distribution in the coefficients. We record the signal-to-noise ratio (SNR) of the resultant image as compared with the original. Second, we test our algorithm without the discriminant function, assuming that the DCT coefficients for all the blocks have a Laplacian distribution with λ determined from the decoder side. Third, we examine the case where the DCT coefficients for all the blocks are assumed to have a Gaussian distribution with σ computed by the decoder. Finally, we test with a mixed model, using the discriminant function described before. We set the threshold $T_0 = 180$. Each block is classified so that we can use the two prior models in decompression.

The results for the two images are summarized in Table 1. In both cases, we observe that using a biased reconstruc-

Table 2 Simulation with different quality factor for Fig. 5.

	SNR for various q for Fig. 5					
	40	50	60	70	80	90
Normal dequantization	19.26 dB	20.40 dB	21.67 dB	23.47 dB	26.25 dB	31.58 dB
Mixed model	19.56 dB	20.74 dB	22.02 dB	23.90 dB	26.67 dB	31.88 dB
Gain	0.30 dB	0.34 dB	0.35 dB	0.43 dB	0.42 dB	0.30 dB

Table 3 Simulation with different quality factor for Fig. 6.

	SNR for various q for Fig. 6					
	40	50	60	70	80	90
Normal dequantization	16.76 dB	17.75 dB	18.85 dB	20.36 dB	22.79 dB	27.61 dB
Mixed model	17.02 dB	18.05 dB	19.17 dB	20.70 dB	23.18 dB	27.97 dB
Gain	0.26 dB	0.30 dB	0.32 dB	0.34 dB	0.39 dB	0.36 dB

tion always produces an image with better quality. For Fig. 5, we see that assuming all the blocks have a Gaussian distribution produces better SNR than a Laplacian distribution. This is in-line with the earlier discussion that for a document with text, the transform coefficient distribution resembles Gaussian. Using a mixed model will further increase the quality by only a small margin. On the other hand, for Fig. 6, which is predominantly image, using a Laplacian model produces better quality output than using a Gaussian model. Again, this is in accordance with the theoretical discussion in Sec. 2. A mixed model is seen to produce little improvement over a single model. These results indicate that if a document has predominantly text or images, a single model will suffice. However, if both have significant proportions, it is better to use the discriminant function to classify the document, and then apply the appropriate model for the best decompression performance. In either case, though, a mixed model is seen to produce at least as good a result as a single model.

Next, we examine the gain due to biased reconstruction for various compression ratios. In most JPEG codecs, there is a quality parameter used to adjust the quantization matrix. The quality factor q ranges from 0 to 100. We adjust the encoding matrix by

$$\tilde{Q}_e = \begin{cases} Q_e \dfrac{50}{q} & \text{for } q \leqslant 50 \\ Q_e(2-0.02q) & \text{for } q > 50 \end{cases}. \qquad (24)$$

So a small q corresponds to worse quality. Using the scaled quantization matrix, we investigate the SNR for quality factors from 40 to 90. The results for the two images are shown in Tables 2 and 3. We see that using a mixed model can achieve consistently around 0.3-dB gain in almost all cases, sometimes even above 0.4-dB gain for quality factors around 70 and 80, which are the usual operating range for JPEG.

6 Conclusions

We propose a mechanism of improving the decompression quality of compound documents by taking advantage of the DCT coefficient distributions. This method is seen to produce documents with better quality than when using the baseline JPEG. However, this is achieved at the expense of more computation. Also, we need a more robust mechanism in estimating the parameters for various distributions. Further improvemen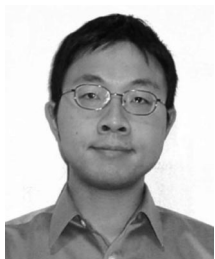t can be made by using the generalized Gaussian distribution as a model for both text and natural images, and by reducing the computational workload in this algorithm.

References

1. I. Witten, A. Moffat, and T. Bell, *Managing Gigabytes: Compressing and Indexing Documents and Images*, 2nd ed., Morgan Kaufmann Publishers, New York (1999).
2. W. Pennebaker and J. Mitchell, *JPEG Still Image Data Compression Standard*, Van Nostrand Reinhold, New York (1992).
3. D. Taubman and M. Marcellin, *JPEG 2000: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic Publishers, Boston (2001).
4. B. G. Haskell, P. G. Howard, Y. A. LeCun, A. Puri, J. Ostermann, M. Reha Civanlar, L. Rabiner, L. Bottou, and P. Haffner, "Image and video coding—emerging standards and beyond," *IEEE Trans. Circuits Syst. Video Technol.* **8**(7), 814–37 (Nov. 1998).
5. K. Konstantinides and D. Tretter, "A JPEG variable quantization method for compound documents," *IEEE Trans. Image Process.* **9**(7), 1282–1287 (Jul. 2000).
6. M. G. Ramos and R. L. de Queiroz, "Classified JPEG coding of mixed document images for printing," *IEEE Trans. Image Process.* **9**(4), 716–720 (Apr. 2000).
7. E. Y. Lam, "Improved mixed document compression by using the DCT coefficient distributions," in *PICS 2003: PICS Conf. Intl. Tech. Conf. Sci. Syst. Digital Photography*, pp. 518–521 (2003).
8. R. Prost, Y. Ding, and A. Baskurt, "JPEG dequantization array for regularized decompression," *IEEE Trans. Image Process.* **6**(6), 883–888 (Jun. 1997).
9. E. Y. Lam and J. W. Goodman, "Discrete cosine transform domain restoration of defocused images," *Appl. Opt.* **37**(26), 6213–6218 (Sep. 1998).
10. J. Price and M. Rabbani, "Biased reconstruction for JPEG decoding," *IEEE Signal Process. Lett.* **6**(12), 297–299 (Dec. 1999).
11. R. Reininger and J. Gibson, "Distributions of the two-dimensional DCT coefficients for images," *IEEE Trans. Commun.* **COM-31**(6), 835–839 (Jun. 1983).
12. F. Müller, "Distribution shape of two-dimensional DCT coefficients of natural images," *Electron. Lett.* **29**(22), 1935–1936 (Oct. 1993).
13. T. Eude, R. Grisel, H. Cherifi, and R. Debrie, "On the distribution of the DCT coefficients," *IEEE Trans. Acoust., Speech, Signal Process.* **5**, 365–368 (1994).
14. S. R. Smoot and L. A. Rowe, "Study of DCT coefficient distributions," *Proc. SPIE* **2657**, 403–411 (1996).
15. E. Y. Lam and J. W. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Trans. Image Process.* **9**(10), 1661–1666 (Oct. 2000).
16. E. Y. Lam, "Improved mixed document compression by using the DCT coefficient distributions," *IEEE Signal Process. Lett.* (in press).
17. A. Leon-Garcia, *Probability and Random Processes for Electrical Engineering*, 2nd ed., Addison Wesley, Reading, MA (1994).
18. J. Rice, *Mathematical Statistics and Data Analysis*, 2nd ed., Duxbury Press, Belmont, CA (1995).

Edmund Y. Lam received the BS degree in 1995, the MS degree in 1996, and the PhD degree in 2000, all in electrical engineering from Stanford University. At Stanford, he was a member of the Information Systems Laboratory, conducting research for the Stanford Programmable Digital Camera project. His focus was on developing image restoration algorithms for digital photography. Outside Stanford, he also consulted for industry in the areas of digital camera system design and algorithm development. Before returning to academia, he worked in the Reticle and Photomask Inspection Division (RAPID) of KLA-Tencor Corporation in San Jose, California, as a senior engineer. His responsibility was to improve on the core die-to-die and die-to-database inspection algorithms, especially for phase shift masks. He is now an assistant professor of electrical and electronic engineering at the University of Hong Kong. His research interests include optics, defect detection, image restoration, and super resolution. He is a member of IEEE and SPIE.