Fifth IFSA World Congress (1993), 1313-1316

Fuzzy Techniques in Optimal Bit Allocation

Seong-Gon Kong

Department of Electrical Engineering
Soongsil University , Seoul, Korea

ABSTRACT

This paper presents a fuzzy system that estimates the optimal bit allocation matrices for the spatially active subimage classes of adaptive transform image coding in noisy channels. Transform image coding is good for image data compression but it requires a transmission error protection scheme to maintain the performance since the channel noise degrades its performance. The fuzzy system provides a simple way of estimating the bit allocation matrices from the optimal bit map computed by the method of minimizing the mean square error between the transform coefficients of the original and the reconstructed images.

1 Introduction

Transform coding is one of popular image data compression techniques in that it shows faithful reconstructed image quality for a given compression ratio, compared to other methods [2]. But the performance of the transform image coding degrades in the presence of channel noise. So transform coding requires a transmission error protection scheme to maintain its performance when we transmit coded transform coefficients through noisy communication channels. Modestino [4] demonstrated that a joint design of the source and channel coders improved reconstructed image quality in noisy channels while maintaining a fixed transmission rate.

As a different approach, Vaishampayan [5] developed an optimal cosine transform coding system in noisy channels by minimizing the mean-square error D between the discrete cosine transform (DCT) coefficients of the original subimages ($X(u,v)$) and of the reconstructed subimages ($\hat{X}(u,v)$) of size $M \times M$:

$$D = \frac{1}{M^2} \sum_{u=0}^{M-1} \sum_{v=0}^{M-1} E[\,|X(u,v) - \hat{X}(u,v)|^2\,] \quad , \quad (1)$$

where $E[\cdot]$ denotes the expectation operator, and $E[\,|X(u,v) - \hat{X}(u,v)|^2\,]$ defines a distortion function. Here the system considers the optimal bit allocation process as an optimization problem.

Adaptive transform image coding systems [1] [3] classify subimages into four spatially active subimage classes according to image activity level measured by the AC energies of the subimage. Each subimage class require a bit allocation matrix to encode its pixels with different bit allocation in order to improve coding performance. This paper presents a fuzzy system that provides a simple way of generating the four optimal bit allocation matrices in noisy channels. The fuzzy system is trained with the input-output product-space data from the given optimal bit allocation process.

2 Stochastic Image Model

We can model an image as a two-dimensional separable and stationary Gauss-Markov random field,

$$\begin{aligned} x(m,n) &= \rho_r x(m-1,n) + \rho_c x(m,n-1) \\ &\quad - \rho_r \rho_c x(m-1,n-1) + w(m,n) \quad (2) \end{aligned}$$

where ρ_r and ρ_c denote the vertical (row) and horizontal (column) correlation coefficients, and $\{w(m,n)\}$ represents a two-dimensional independent and identically distributed Gaussian random sequence with zero mean and variance σ_w^2. If $\{x(m,n)\}$ has zero mean and variance σ_x^2, then

$$
\begin{aligned}
E[x^2(m,n)] &= \\
&\sigma_x^2(\rho_r^2+\rho_c^2+\rho_r^2\rho_c^2) + 2\rho_r\rho_c E[x(m-1,n)x(m,n-1)] \\
&- 2\rho_r^2\rho_c E[x(m-1,n)x(m-1,n-1)] \\
&- 2\rho_r\rho_c^2 E[x(m,n-1)x(m-1,n-1)] + \sigma_w^2 \qquad (3) \\
&= \sigma_x^2(\rho_r^2+\rho_c^2+\rho_r^2\rho_c^2+2\rho_r^2\rho_c^2-2\rho_r^2\rho_c^2-2\rho_r^2\rho_c^2) + \sigma_w^2 \qquad (4) \\
&= \sigma_x^2(\rho_r^2+\rho_c^2-\rho_r^2\rho_c^2) + \sigma_w^2 \qquad (5)
\end{aligned}
$$

since

$$
\rho_r = \frac{R(1,0)}{R(0,0)} = \frac{E[x(m,n)x(m-1,n)]}{E[x^2(m,n)]} \qquad (6)
$$

$$
\rho_c = \frac{R(0,1)}{R(0,0)} = \frac{E[x(m,n)x(m,n-1)]}{E[x^2(m,n)]} \qquad (7)
$$

The autocorrelation function $R(k,l)$ can be approximated by the sample autocorrelation functions. Therefore the noise variance can be computed as

$$
\sigma_w^2 = \sigma_x^2(1-\rho_r^2)(1-\rho_c^2) \qquad (8)
$$

For the Gauss-Markov image model defined in (2), the variance $\sigma^2(u,v)$ of the DCT coefficient $X(u,v)$ factors as

$$
\sigma^2(u,v) = \sigma_c^2(u)\sigma_r^2(v) \qquad (9)
$$

We can easily compute $\sigma_c^2(u)$ and $\sigma_r^2(v)$ using the two-dimensional DCT algorithm as

$$
\begin{aligned}
\sigma_c^2(u) &= \\
&\frac{2\sigma_x}{M}a^2(u)\sum_{m=0}^{M-1}\sum_{n=0}^{M-1}\rho_c^{|m-n|}\cos\frac{(2m+1)u\pi}{2M} \\
&\cos\frac{(2n+1)u\pi}{2M} \qquad (10)
\end{aligned}
$$

$$
\begin{aligned}
\sigma_r^2(v) &= \\
&\frac{2\sigma_x}{M}a^2(v)\sum_{m=0}^{M-1}\sum_{n=0}^{M-1}\rho_r^{|m-n|}\cos\frac{(2m+1)v\pi}{2M} \\
&\cos\frac{(2n+1)v\pi}{2M} \qquad (11)
\end{aligned}
$$

where $a(0) = 1/\sqrt{2}$ and $a(k) = 1$ for $k = 1,\ldots,M-1$. Equations (10) and (11) correspond to the diagonal elements of the $M \times M$ DCT coefficients of the matrix $\rho^{|m-n|}$, $0 \le m,n \le M-1$, multipled by σ_x.

In the adaptive transform image coding, we divide an image into 16×16 subimages, and classify the subimages into four spatially active subimage classes. Each subimage class assumes the Gauss-Markov image model with different parameter values. Table 1 shows the parameters of each subimage class computed from the two sample images, the Lenna and the F-16 jet fighter. All the parameters ρ_r, ρ_c, mean, and σ_x^2 are averaged over 64 subimages for each subimage class. Subimage class 1 contains lots of image details and therefore shows lower correlation values, Class 4 includes less image details, so it shows higher correlation.

Image	Class	ρ_r	ρ_c	σ_x^2	mean
Lenna	1	0.8886	0.8348	2288.01	85.79
	2	0.9123	0.8835	1302.38	91.70
	3	0.9322	0.9248	447.45	109.21
	4	0.9351	0.9343	107.33	109.31
F-16	1	0.8990	0.9021	3447.05	139.42
	2	0.9279	0.9266	1789.25	165.00
	3	0.9361	0.9368	198.11	194.58
	4	0.9377	0.9374	12.58	211.30

Table 1: Parameters computed from the Lenna and the F-16 images.

8	8	8	8	8	8	8	8	8	8	8	8	8	7	7	7	7	7	7	6	6	6	6	6	6	6	6	6	6	6	6	6
8	8	8	8	8	7	7	6	6	5	5	5	4	3	3	3	3	3	3	3	3	3	3	3	3	2	2	2	2	2	2	2
8	8	8	7	6	5	5	3	3	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1
8	8	7	6	5	3	3	3	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	8	6	5	3	3	2	2	2	2	2	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
8	7	5	3	3	2	2	2	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	7	5	3	2	2	2	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	6	3	3	2	2	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	6	3	2	2	2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	5	3	2	2	1	1	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
8	5	3	2	2	1	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
8	5	3	2	1	1	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
8	4	2	2	1	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
8	4	2	2	1	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
7	3	2	1	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
7	3	2	1	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
7	3	2	1	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
7	3	2	1	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
7	3	2	1	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
7	3	2	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
7	3	2	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
7	3	2	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
7	3	2	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
6	3	2	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
6	3	2	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
6	3	1	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
6	3	1	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
6	2	1	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
6	2	1	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
6	2	1	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
6	2	1	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
6	2	1	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

Figure 1: The 32×32 optimal bit map for channel noise with $\epsilon = 0.01$.

3 Optimal Bit Allocation for Subimage Classes

The optimal bit-allocation process in [5] provides the input-output product-space data of the form $(\sigma; r)$. Here $\sigma(u,v)$ denotes the standard deviation of the DCT coefficient $X(u,v)$ of the Gauss-Markov image model. The given 32×32 standard deviation matrix $\sigma(u,v)$ are computed by Equation (9) with parameters $\rho_r = 0.9790$, $\rho_c = 0.9746$, mean $= 73.57$, and $\sigma_x^2 = 1816.56$ [5] for the Lenna image. $r(u,v)$ represents the (u,v)th element of the 32×32 optimal bit allocation matrix shown in Figure 1 computed for noisy binary symmetric channel with bit error probability $\epsilon = 0.01$ and the bit rate of

1 bit/pixel. The value of $r(u,v)$ indicates the bit allocation of the transform coefficient $X(u,v)$ and is either $0,1,\dots,r_{max}$. We often choose the maximum value of r_{max} as 8. The standard deviation and the optimal bit allocation gave 1024 ($= 32 \times 32$) input-output product-space data of the form (σ, r).

A fuzzy system estimated the four optimal bit allocation matrices for each subimage class of the adaptive cosine transform coding. A fuzzy rule $(\sigma_k(u,v); r_k(u,v))$ represents the association:

IF $\sigma_k(u,v)$ belongs to I_i, THEN assign i to $r_k(u,v)$.

where σ_k and r_k denote the standard deviation and bit allocation for the kth subimage class with $k = 1,\dots,4$ and $u,v = 0,1,\dots,M-1$. I_i represents fuzzy decision intervals, $i = 0,1,\dots,r_{max}$. So $(r_{max}+1)$ fuzzy rules generated the four optimal bit allocation matrices for each spatially active subimage class.

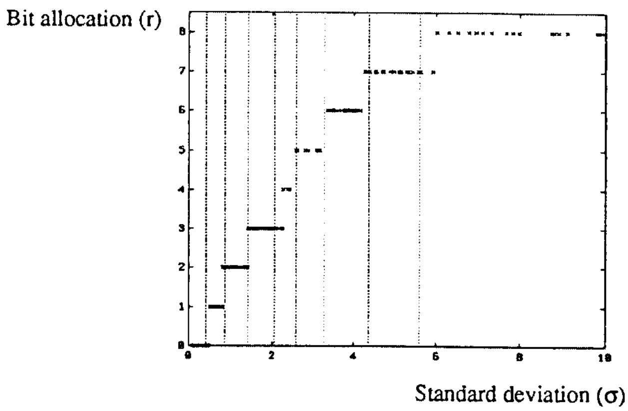

Figure 2: Training data $(\sigma; r)$ and the decision intervals.

We can estimate the decision interval I_i from the training data from the input-output product-space $(\sigma; r)$. Let q_i denote the standard deviation value $\sigma(u,v)$ averaged over each bit allocation value i,

$$q_i = \frac{1}{N_i} \sum_{\{(u,v): r(u,v)=i\}} \sigma(u,v) \qquad (12)$$

where $i = 0,1,\dots,r_{max}$ and N_i denotes the number of $\sigma(u,v)$ corresponding to the bit allocation $r(u,v) = i$.

The fuzzy cells can be defined for the decision intervals I_i as non-overlapping intervals of membership functions, $[p_i,\ p_{i+1}]$, $i = 0,1,\dots,r_{max}$, where

$$p_i = c\left(\frac{q_{i-1}+q_i}{2}\right) \qquad (13)$$

$i = 1,\dots,r_{max}-1$, and $p_0 = 0$, $p_{r_{max}} = q_{r_{max}-1} + |q_{r_{max}-1} - p_{r_{max}-1}|$, and $p_{r_{max}+1} = \infty$. For example, c takes 2.4 for 8:1 compression and 4.6 for 16:1 compression.

The input-output product-space data provided $p_1 = 3.44$, $p_2 = 5.66$, $p_3 = 9.30$, $p_4 = 13.46$, $p_5 = 16.97$, $p_6 = 21.46$, $p_7 = 28.49$, and $p_8 = 36.55$. Figure 2 plots the training data $(\sigma; r)$ with decision intervals I_i.

The fuzzy system generated the four bit allocation matrices for the decision intervals I_i "trained" from the input-output product-space data. Figure 3 shows the four optimal bit allocation matrices for approximately 8:1 compression computed by the fuzzy system with the parameters given in Table 1. Table 2 demonstrates the coding performance of the adaptive transform image coding techniques by the fuzzy subimage classification [3] and by that of Chen system [1] using the four bit allocation matrices. The fuzzy system showed similar but slightly better signal-to-noise ratio while maintaining approximately 8:1 and 16:1 compression ratios.

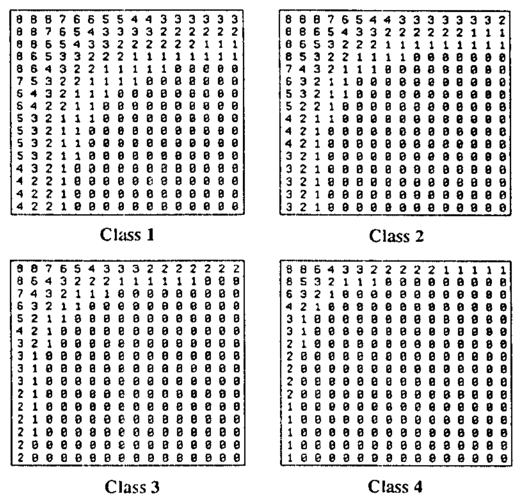

Figure 3: Optimal bit allocation matrices in noisy channels for approximately 8:1 compression computed by the proposed fuzzy system with the parameters given in Table 1.

The number of arithmetic operations provided a rough measure of computational complexity for the fuzzy system. The fuzzy system required less computational efforts for to estimate the optimal bit allocation matrices than Vaishampayan's bit allocation process, which required heavy computation in order to compute and minimize the complex distortion function $E[|X(u,v) - \hat{X}(u,v)|^2]$.

Image		SNR(dB)	C_r	SNR(dB)	C_r
Lenna	Fuzzy	27.97	8.2	25.62	16.7
	Chen	27.93	8.3	25.56	16.9
Jet	Fuzzy	27.64	8.2	24.85	16.8
	Chen	27.37	8.3	24.62	16.9

Table 2: Coding performance of the adaptive transform image coding techniques by the fuzzy and Chen subimage classification methods using the four bit allocation matrices. C_r denotes the compression ratio.

4 Conclusion

Adaptive transform image coding required the four optimal bit allocation matrices to maintain its performance in noisy channels. To compute the optimal bit allocation matrices in noisy channels, Vaishampayan [5] considers the problem as an optimization. The technique is computationally heavy since it should minimize the complex distortion function, the mean-square error of the source and the received transform coefficients.

The fuzzy system, trained with the input-output product-space data from the given optimal bit allocation matrix computed by the optimization technique, estimated the optimal bit allocation matrices for the four subimage classes. The bit allocation matrices by the fuzzy system provided very close coding performance to that of the optimal bit allocation scheme, but with much less computation.

References

[1] W.-H. Chen and C. H. Smith, "Adaptive Coding of Monochrome and Color Images," *IEEE Transactions on Communications*, Vol. 25, No. 11, 1285-1292, November 1977.

[2] A. K. Jain, "Image Data Compression: A Review," *Proceedings of the IEEE,* Vol. 69, No. 3, 349-389, March 1981.

[3] S.-G. Kong and B. Kosko, "Adaptive Fuzzy System for Transform Image Coding," *Proceedings of the International Joint Conference on Neural Networks (IJCNN-91)*, Vol. I, 609-614, July 1991.

[4] Modestino, J. W., Daut, D. G., and Vickers, A. L., "Combined Source-Channel Coding of Images using the Block Cosine Transform," *IEEE Transactions on Communications*, vol. COM-29, 1261-1274, September 1981.

[5] Vaishampayan, V. A. and Farvadin, N., "Optimal Block Cosine Transform Image Coding for Noisy Channels," *IEEE Transactions on Communications*, vol. COM-38, no. 3, 327-336, March 1990.