Journal of Electronic Imaging 1(3), 328–338 (July 1992).

Human visual sensitivity-weighted progressive image transmission using the lapped orthogonal transform

Ricardo L. de Queiroz
K. R. Rao
University of Texas at Arlington
Electrical Engineering Department
P.O. Box 19016
Arlington, Texas 76019

Abstract. *Progressive transmission of images based on the lapped orthogonal transform (LOT), adaptive classification, and human visual sensitivity (HVS) weighting is proposed. HVS weighting for LOT basis functions is developed. This technique is quite general and can be applied to any orthogonal transform. The method is compared with discrete cosine transform (DCT)-based progressive image transmission (PIT). It is shown that LOT-based PIT yields subjectively improved images compared to those based on DCT. This is consistent with the reduction in block structure characteristic of LOT image coding.*

1 Introduction

While progressive image transmission[1] (PIT) can be classified into two major categories, i.e., (1) spatial or pel domain and (2) transform or spectral domain, the latter has gained wide acceptance.[2–10] This is not only due to various adaptive features such as classification,[11–16] spectral selection,[4,7,8] and human visual system (HVS) weighting,[2,7,8,17–21] etc., which can be easily incorporated into the transform coding scheme, but is also due to the VLSI development of coding operations such as transform, quantization, and variable length coding. In addition, PIT based on the discrete cosine transform (DCT) has been extensively investigated. For example, the JPEG (Joint Photographic Experts Group) algorithm[7,8] for the baseline system is DCT based and various hardware/software systems have already been developed for this algorithm. Also, the nonhierarchical extended system of JPEG (both spectral selection and successive approximation) is DCT based. At low bit rates, however, DCT introduces block structure in the reconstructed images.[2] One technique used to reduce or eliminate this artifact is to replace DCT by the lapped orthogonal transform (LOT),[22–28] whose basis vectors overlap across traditional block boundaries. Also because LOT has good filtering properties, it has been applied to compatible coding,[29,30] i.e., coding of the original image/sequence at different spatial resolutions. It has also been combined with vector quantization (VQ) to achieve additional compression.[31] It is intuitively felt that LOT-based PIT should yield subjectively more pleasing pictures compared to the DCT—even during the initial stages. This is the objective of this paper: to develop a LOT-PIT incorporating various adaptive features and to compare it with the DCT-dependent PIT.

In Sec. 2, we will address the Chen-Smith coder, giving a brief summary of the algorithm steps and explaining the incorporation of PIT techniques in this algorithm. Section 3 is reserved for a discussion about the HVS model in the transform domain. Simulations and coder details are presented in Sec. 4, with conclusions given in Sec. 5.

2 PIT with the Chen-Smith Coder

The Chen-Smith coder[12] is based on the zonal sampling strategy. First, the image undergoes an orthogonal transform. The transform coefficients are stored in a buffer and some statistics are computed prior to the decision-making process of (1) which coefficients are transmitted, (2) how these coefficients are quantized, and (3) the order of transmission. We will assume the image has $N \times N$ picture elements (pixels or pels).

The encoding steps can be briefly described as follows:

Paper 92-018 received April 7, 1992; revised manuscript received July 13, 1992; accepted for publication July 16, 1992.
1017–9909/92/$2.00.

- Transform the image using blocks of $M \times M$ pels. Let $N_B = (N/M)^2$ be the total number of blocks in the image. To simplify the presentation, we will use a lexicographic ordering that can obey row or column arrangement. The blocks are then labeled from 1 to N_B. Each one contains M^2 coefficients given as $x_i(u,v)$ for $i = 1, \ldots, N_B$ and $(u,v) \in \{(0,0), \Psi\}$, where Ψ is defined as the set of $M^2 - 1$ block-index pairs, excluding the pair (0,0), as $\Psi \equiv \{(0,1), (0,2), \ldots, (0, M-1), (1,0), (1,1), \ldots, (M-1, M-1)\}$.
- Quantize and code separately the coefficients $x_i(0,0)$ (the dc coefficients) using uniform quantizers.
- Compute the ac energy of each block E_i as

$$E_i = \sum_{(m,n) \in \Psi} x_i^2(m,n) \quad . \tag{1}$$

Sort the energies, and classify the blocks (in sorted order) into N_C equally populated classes.[12] Thus, there would be N_B/N_C blocks in each class. Construct the *class map* $C(i)$ with the classification of each block, where $C(i)$ indicates the class to which the i'th block belongs and is ordered in the original nonsorted sequence. If the i'th block belongs to the class k ($k = 1, \ldots, N_C$), then $C(i) = k$.

- For all blocks belonging to the same class, compute the variances of the transform coefficients and then their standard deviations. Construct N_C *standard deviation maps* with the standard deviations of the coefficients, which are obtained from

$$\sigma_k^2(m,n) = \sum_{i=1}^{N_B} \delta[C(i) - k] x_i^2(m,n) \quad (m,n) \in \Psi \quad , \tag{2}$$

where δ is the Kronecker delta function.

- Merge all N_C standard deviation maps and decide the bit allocation. Based on the rate-distortion theory, we shall iteratively find a distortion value D and a set of integers $B_k(m,n)$ [for $(m,n) \in \Psi$ and $1 \leq k \leq N_C$], so that

$$B_k(m,n) = \tfrac{1}{2} \log_2[\sigma_k^2(m,n)] - \log_2(D) \tag{3}$$

is satisfied given the constraints

$$\sum_{k=1}^{N_C - 1} \sum_{(m,n) \in \Psi} B_k(m,n) = (RN^2 - B_{ov}) \frac{N_C}{N_B} \quad , \tag{4}$$

$$0 \leq B_k(m,n) \leq B_{max} \quad , \tag{5}$$

where B_{max} is the maximum number of bits allowed, B_{ov} is the number of bits required for the transmission of the overhead information, and R is the bit rate in bits/pel for the whole image. Create N_C bit-allocation maps with a one-to-one correspondence with the elements of the standard deviation maps.

- Reestimate the standard deviations using the bit-allocation maps:

$$\hat{\sigma}_k(m,n) = c 2^{B_k(m,n) - 1} \quad 1 \leq k \leq N_C \quad (m,n) \in \Psi \quad , \tag{6}$$

where c is a normalization factor. Reference 12 suggested that c be chosen as the maximum $\sigma_k(m,n)$ for which $B_k(m,n) = 1$ to avoid excessive clipping.

- Send class map c and the bit-allocation maps as side information.
- Quantize, encode, and send all the coefficients, using the reestimated variances. A coefficient $x_i(m,n)$ (block i), which belongs to class $k[C(i) = k]$, is scaled [divided by $\hat{\sigma}_k(m,n)$], applied to a quantizer with $2^{B_k(m,n)}$ levels, and encoded with $B_k(m,n)$ bits. If $B_k(m,n) = 0$, the particular coefficient is not transmitted.

The receiver may first decode the side information and the dc coefficients. Given the class map, the bit-allocation maps, and the normalization factor c, the decoder can reconstruct the standard deviations used to scale the quantizers as in Eq. (6). With the maps reconstructed, and with the knowledge of the transmission order, the decoder can exactly determine the position of the incoming coefficient, the class of its block, how many bits were assigned to it, and the variance used for quantization. Therefore, the receiver can decode the coefficients, apply an inverse transform, and obtain the image.

The overhead is made by the class map, the bit-allocation maps, and by c. Quantizing c with 16 bits, the total amount of overhead is given by:

$$B_{ov} = N_B \log_2(N_C) + N_C(M^2 - 1) \times \log_2(B_{max} + 1) + 16 \quad . \tag{7}$$

If $M = 8$, $N = 256$, $N_C = 8$, $B_{max} = 7$, then $B_{ov} = 4552$, which is equivalent to an approximate rate of 0.07 bit/pel, requiring about 2 s of transmission on a 2400 bits/s communication rate.

To use PIT, we transmit data in the following order: (1) dc coefficients in any predefined order, (2) class map c and bit-allocation maps, (3) ac coefficients. The transmission of the ac coefficients[2] is made by spanning the blocks and sending first the elements $x_i(m,n)$, which would yield a higher contribution to the reconstructed image. To minimize the reconstruction error, we send the coefficients with higher variances. Alternatively, we can incorporate some information about the spatial response of the visual system, by using weighted standard deviations. If one assumes that the estimated standard deviation is a good measure of the real standard deviation of a particular coefficient (at least, is the best information we have at hand), the priority can be decided based on the weighting of the standard deviation maps by a matrix $\mathbf{H}(m,n)$ containing spatial information about the HVS. Let

$$\eta_k(m,n) = \hat{\sigma}_k(m,n) \mathbf{H}(m,n) \; ; \quad 1 \leq k \leq N_C; \quad (m,n) \in \Psi \quad . \tag{8}$$

The order for transmission of the coefficients is then defined by sending first the coefficients $[x_i(m,n); C(i) = k]$, which correspond to: (1) greater value of $\eta_k(m,n)$; (2) if two or more $\eta_k(m,n)$ have the same value, take the one with smaller value of $m+n$; or (3) if there is still any ambiguity, take the smaller value of k.

The first item is the only one that follows any theoretical explanation; the last two are included merely for eliminating ambiguities, such as two equal values, and can be changed without affecting the performance. Note that using Eq. (6),

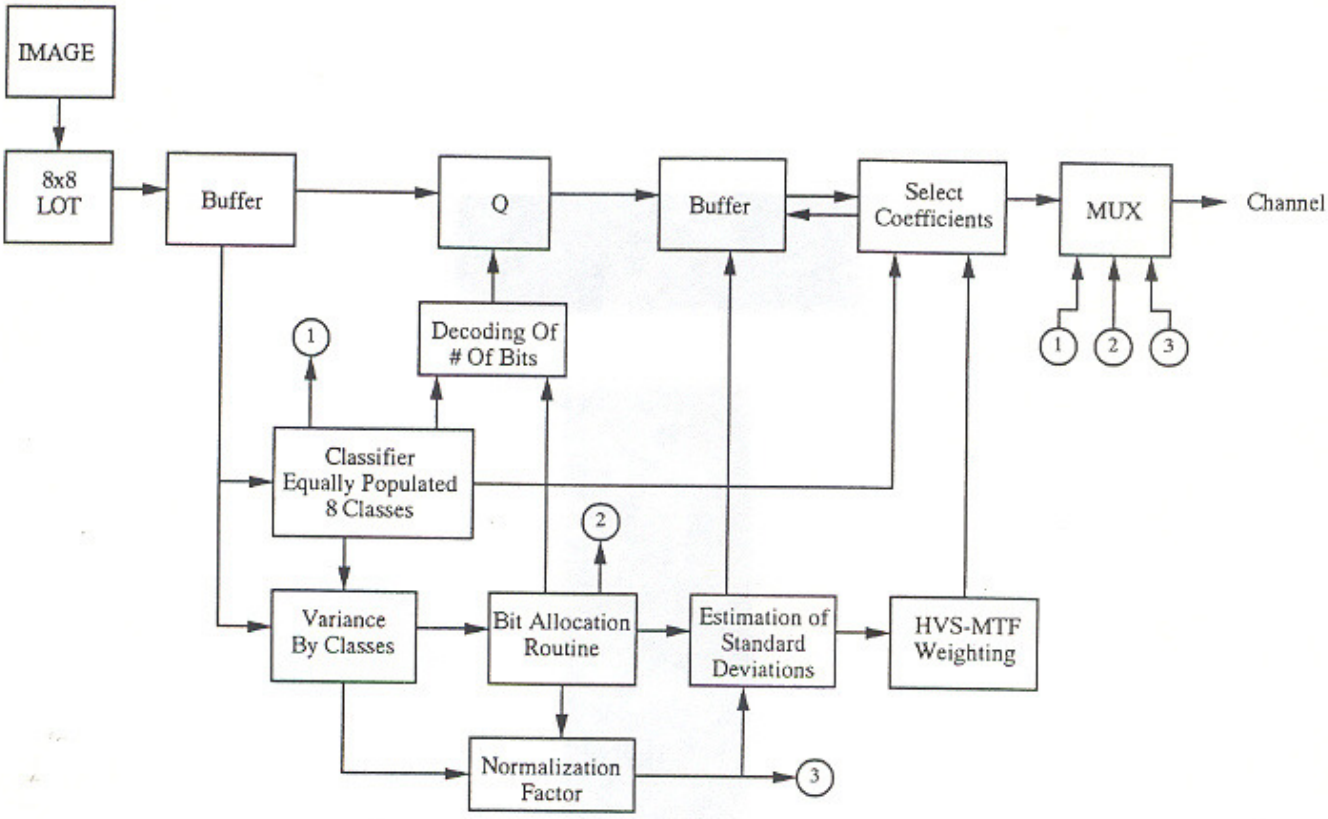


Fig. 1 Coder diagram for PIT using LOT.

we can take the log and sum $\log(2/c)$ on both sides of Eq. (8) so that $\eta_k(m,n)$ can be redefined as

$$\eta_k(m,n) = B_k(m,n) + \log_2[\mathbf{H}(m,n)] \quad . \tag{9}$$

Although having a different value, this representation still maintains the transmission order, addressing directly the bit-allocation maps. As long as both encoder and decoder have the same maps and use the same weighting matrix, there will be no overhead for indicating the transmission order.

The coder has some limitations. First, the maximum number of different variances used for scaling the quantizers is $B_{\max}$. For high rates (>4 bits/pel), the performance decreases, since there will no longer be coefficients with only a few bits allocated. Second, it is not possible to apply HVS weighting to quantization without causing excessive mismatch or amplification of distortion because of the reestimation procedure in Eq. (6). It can be overcome by the transmission of standard deviations in place of the bit-allocation maps. We are interested in "small" pictures, such as 256×256 pel images. For these types of images, using 8 or 16 classes, the overhead for fully transmitting the variance maps would be prohibitive. The performance of this coder can be improved in several ways. For example, by choosing the proper parameters (block size, number of classes, and bit rate), the coder can achieve very good performance. The great advantage of the Chen-Smith approach is that it is quite insensitive to the transform used. One can interchangeably use DCT, LOT, extended lapped transforms,[27] or any transform resulting in blocks of $M \times M$ coefficients without any alteration in the algorithm (except for the weighting matrix and, possibly, coding details). This is the main reason for choosing the Chen-Smith coder.

The coder and decoder block diagrams employing the LOT are presented in Figs. 1 and 2, respectively.

3 The HVS Weighting Matrix

A complete study of the psychophysical properties of the visual system is well beyond the scope of this paper. Our intention is restricted to the determination of a spatial response weighting matrix for use with the LOT coefficients. We now present a procedure that allows us to find a HVS weighting function for any transform.

Reference 2 discussed the application of a linear function describing the HVS to spatial variations. Although the HVS model response is not linear, this principle was used with good results and further discussion on the subject is left to Ref. 2. Given a linear transfer function representing the unidimensional spatial HVS as $H(f)$ (where f is given in cycles per degree of the visual angle subtended), we will assume this model to be reliable and it will serve as the basis for the rest of this section. However, we will present our results as a function of the model in order to allow one to change $H(f)$ if desired. Further, the usual assumptions follow:

- The screen has a 1:1 ratio and has uniform brightness when displaying a uniform image.
- The viewer is situated at a distance v from the screen, right in front of its geometric center.

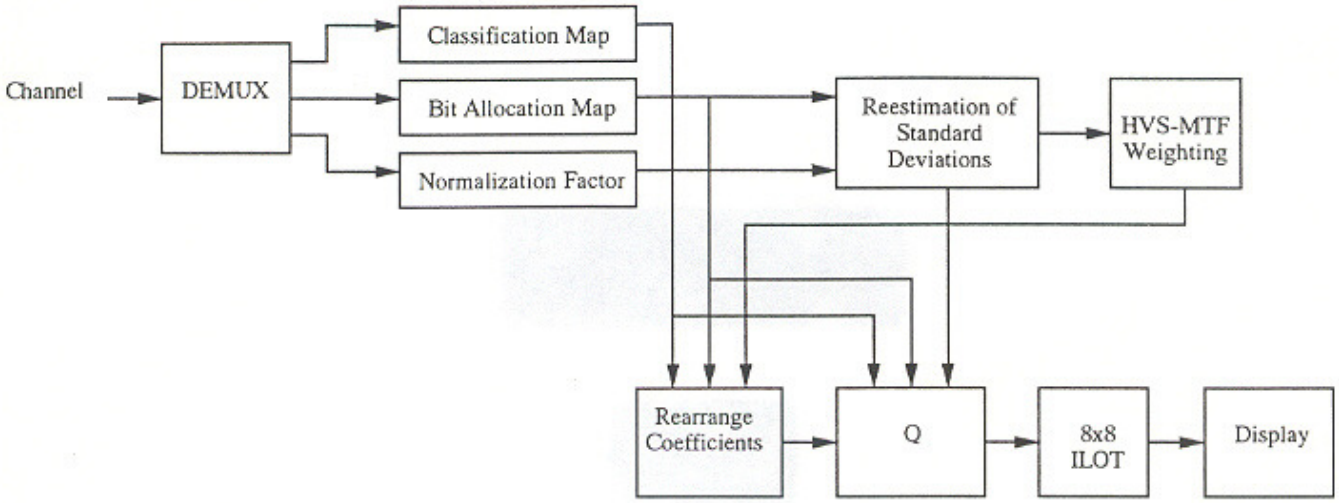


Fig. 2 Decoder diagram for PIT using LOT.

- The screen has width w and each row (column) has N pels.
- The viewer can observe approximately the same density of pels-per-degree (spatial) in any region of the screen.

Let α be the ratio of viewer distance (v) by screen width (w), i.e., $\alpha \equiv v/w$. This factor is the relative distance of the observer. The maximum visible frequency in cycles per degree is obtained when the discrete signal displayed has its maximum frequency component, which is half of the sampling frequency. In other words, in N samples it is possible to observe $N/2$ cycles. The maximum visible frequency can be found as:

$$f_{\max} = \frac{N/2}{2\theta} = \frac{N}{4 \arctan\left(\frac{1}{2\alpha}\right)} \text{ cycles/degree} , \tag{10}$$

where θ in degrees is the viewing angle, from the center to the extreme of the screen, and $\tan(\theta) = w/2v = 1/2\alpha$. We, therefore, can represent a discrete sensitivity function as

$$H_D(e^{j\omega}) = H_D(e^{j2\pi f}) = H(f/f_{\max}) ; \quad |f| < f_{\max} . \tag{11}$$

An orthogonal block transform is a special case of a lapped transform in which there are as many basis functions as elements in each basis function.[26] Furthermore, lapped transforms are equivalent to paraunitary filter banks.[26] Therefore, we can always regard any discrete, real, and orthogonal (lapped or block) transform as a filter bank.[26,32,33] The analysis filters' coefficients are the time-reversed basis functions elements.[26,32] Suppose the M basis functions have elements $p_k(n)$ ($k = 0, 1, \ldots, M-1$ and $n = 0, 1, \ldots, L-1$). The equivalent analysis filter bank is shown in Fig. 3, where each filter [with coefficient $f_k(n)$] is equal to a basis function of the LOT, i.e., $f_k(n) = p_k(L-1-n)$ for $n = 0, 1, \ldots, L-1$. For the particular case of the LOT of M bands, $L = 2M$, but for the DCT we have $L = M$ (as any block transform). In Fig. 3, with $x(n)$ as the input signal to the filter bank,

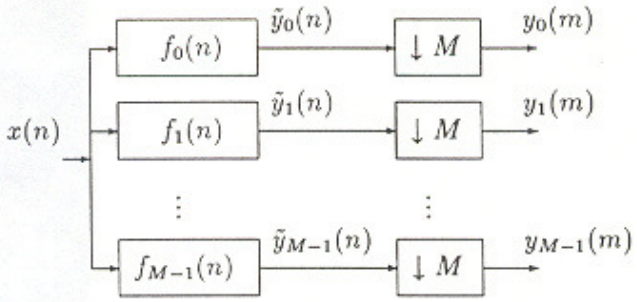


Fig. 3 Analysis section of a critically decimated M-band filter bank where $x(n)$ is the input signal and $\tilde{y}_k(mM)$ are the subband signals after filtering ($0 \le k \le M-1$). The subband signals are decimated resulting in $y_k(m) = \tilde{y}_k(mM)$. The filters' impulse responses $f_k(n)$ are the time-reversed basis functions of the transform.

$\tilde{y}_k(n)$ corresponds to each subband (filtered signals), and $y_k(n)$ is the subband signal after decimation. Let $F_k(e^{j\omega})$ be the frequency response of $f_k(n)$. Figure 4 shows the frequency response of the first three filters (basis functions) for a one-dimensional LOT with 8 bands (i.e., a 16×8 LOT matrix). Similar results for the DCT are found in Fig. 5. The same procedure can also be applied to nonuniform filter banks such as those resulting from the use of hierarchical structures. If, in Fig. 3, the input $x(n)$ has a power spectral density (psd) given by $S_x(\omega)$, and denoting the PSD of $\tilde{y}_k(n)$ and $y_k(n)$ as $S_{\tilde{y}k}(\omega)$ and $S_{yk}(\omega)$, we have:

$$S_{\tilde{y}k}(\omega) = S_x(\omega) |F_k(e^{j\omega})|^2 . \tag{12}$$

After the decimator, $y_k(n) = \tilde{y}_k(nM)$, and

$$S_{yk}(\omega) = \sum_{r=0}^{M-1} S_{\tilde{y}k}\left(\frac{\omega - 2\pi r}{M}\right) . \tag{13}$$

As

$$\int_a^b S_{\tilde{y}k}(\omega)\, d\omega = \int_{2\pi-b}^{2\pi-a} S_{\tilde{y}k}(\omega)\, d\omega ,$$

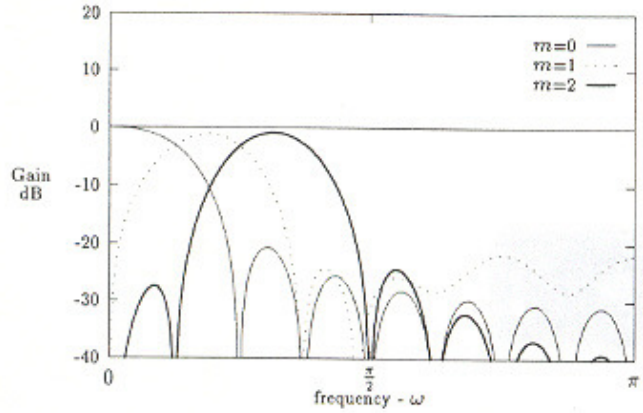


Fig. 4 Frequency response in decibels of the filters $f_m(n)$ corresponding to the first three basis functions of the LOT, i.e., $|F_m(e^{j\omega})|$, $m=0, 1, 2$.

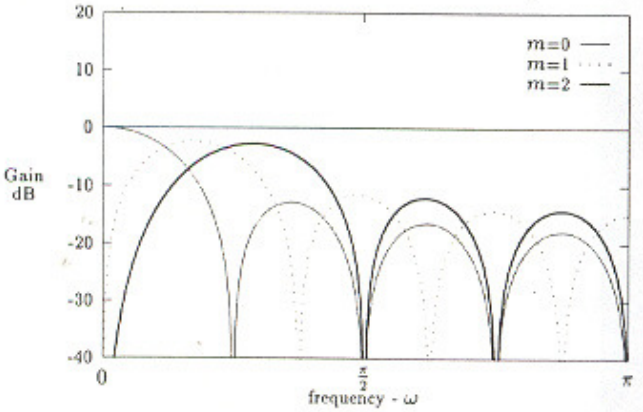


Fig. 5 Frequency response in decibels of the filters $f_m(n)$ corresponding to the first three basis functions of the DCT, i.e., $|F_m(e^{j\omega})|$, $m=0, 1, 2$.

the variance of y_k is given by

$$\zeta_k^2=\frac{1}{\pi}\int_0^{\pi} S_{y_k}(\omega)\,d\omega=\frac{1}{\pi}\int_0^{\pi} S_{\bar{y}_k}(\omega)\,d\omega \quad . \tag{14}$$

Alternatively, this result could be shown using the fact that if $u(n)$ is a stationary process, then $\text{var}[u(n)]=\text{var}[u(Mn)]$. Therefore, $\text{var}[\bar{y}_k(n)]=\text{var}[y_k(n)]$ and the preceding equation is also true.

Roughly, if a signal is filtered by $H_D(e^{j\omega})$, the signal and its filtered version would be indistinguishable for the observer to whom $H_D(e^{j\omega})$ is a perfect sensitivity model. If this signal has a flat PSD (white noise), the filtered signal has the PSD shaped by the filter, letting one know the relative importance of each frequency component for the observer. If this colored signal is split into subbands, as when using the LOT, how can we measure the importance of each subband component? A sampling in the frequency domain would be imprecise and very dependent on the phase of the sampling train, since there would be only M bands of width π/M. This bandwidth can be large enough to allow significant variations of the input PSD. Since we are measuring up to the second-order statistics in the image, and on those we may apply the weighting matrix, one possible

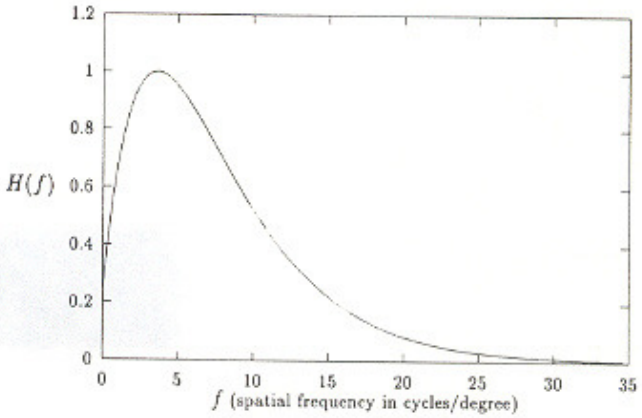


Fig. 6 HVS model function used in this paper.[2]

solution would be the measure of the variance of each band. These variances can provide the relative significance of each subband. Note that as M increases, L increases, and the filters are becoming close to ideal filters and the bandwidth is becoming narrower. In the limit, the approximations by sampling and by variance computation would yield the same results.

If a white noise with unit variance is input to the linear system $H_D(e^{j\omega})$, and its output is transformed using the LOT, then Eq. (14) is given by:

$$\zeta_k^2=\frac{1}{\pi}\int_0^{\pi} |H_D(e^{j\omega})|^2\,|F_k(e^{j\omega})|^2\,d\omega \tag{15}$$

The continuous HVS model function as used in Ref. 2 is plotted in Fig. 6. As previously stated, the frequency f is given in cycles per degree of visual angle subtended. The model is given by:

$$H(f)=2.46(0.1+0.25f)e^{-0.25f} \quad . \tag{16}$$

The corresponding weights ζ_k can be found using Eqs. (11) and (15).

The two-dimensional case is just an extension of these results, since the transform is separable. We are interested in weights ζ_{ij}, $(i,j)\in\Psi$, which can be derived from

$$\zeta_{ij}^2=\frac{1}{\pi^2}\int_0^{\pi}\int_0^{\pi} |H_D(e^{j\omega_1}, e^{j\omega_2})|^2 \times|F_{ij}(e^{j\omega_1}, e^{j\omega_2})|^2\,d\omega_1\,d\omega_2 \quad , \tag{17}$$

where

$$H_D(e^{j\omega_1}, e^{j\omega_2})=H_D(e^{j2\pi f_1}, e^{j2\pi f_2})=H(f_p/f_{\max}) \tag{18}$$

and

$$f_p=\sqrt{f_1^2+f_2^2}\;;\quad |f_1|<f_{\max}\;,\quad |f_2|<f_{\max}$$

and

$$F_{ij}(e^{j\omega_1}, e^{j\omega_2})=F_i(e^{j\omega_1})\,F_j(e^{j\omega_2}) \quad . \tag{19}$$

In our application, we are weighting standard deviation values and we use ζ_{ij} instead of the squared value. Figure 7

0.6854	0.8698	0.9883	1.0000	0.9546	0.8703	0.7706	0.6793
0.8698	0.9371	0.9930	0.9821	0.9294	0.8457	0.7475	0.6598
0.9883	0.9930	0.9963	0.9606	0.8987	0.8154	0.7194	0.6362
1.0000	0.9821	0.9606	0.9114	0.8458	0.7659	0.6752	0.5984
0.9546	0.9294	0.8987	0.8458	0.7816	0.7073	0.6241	0.5543
0.8703	0.8457	0.8154	0.7659	0.7073	0.6409	0.5667	0.5047
0.7706	0.7475	0.7194	0.6752	0.6241	0.5667	0.5028	0.4493
0.6793	0.6598	0.6362	0.5984	0.5543	0.5047	0.4493	0.4024

(a) $\alpha = 4$; $f_{max} = 9$ cycles/degree

0.7460	0.9223	1.0000	0.9542	0.8566	0.7341	0.6071	0.5101
0.9223	0.9686	0.9836	0.9214	0.8222	0.7051	0.5829	0.4911
1.0000	0.9836	0.9515	0.8742	0.7749	0.6653	0.5503	0.4655
0.9542	0.9214	0.8742	0.7955	0.7032	0.6051	0.5021	0.4265
0.8566	0.8222	0.7749	0.7032	0.6222	0.5375	0.4483	0.3824
0.7341	0.7051	0.6653	0.6051	0.5375	0.4665	0.3916	0.3356
0.6071	0.5829	0.5503	0.5021	0.4483	0.3916	0.3312	0.2854
0.5101	0.4911	0.4655	0.4265	0.3824	0.3356	0.2854	0.2468

(b) $\alpha = 5$; $f_{max} = 11.2$ cycles/degree

0.8090	0.9702	1.0000	0.8988	0.7576	0.6112	0.4710	0.3804
0.9702	0.9920	0.9627	0.8533	0.7171	0.5803	0.4476	0.3630
1.0000	0.9627	0.8966	0.7846	0.6583	0.5354	0.4145	0.3379
0.8988	0.8533	0.7846	0.6845	0.5759	0.4714	0.3676	0.3012
0.7576	0.7171	0.6583	0.5759	0.4877	0.4024	0.3169	0.2610
0.6112	0.5803	0.5354	0.4714	0.4024	0.3348	0.2664	0.2206
0.4710	0.4476	0.4145	0.3676	0.3169	0.2664	0.2146	0.1789
0.3804	0.3630	0.3379	0.3012	0.2610	0.2206	0.1789	0.1498

(c) $\alpha = 6$; $f_{max} = 13.4$ cycles/degree

0.8629	1.0000	0.9750	0.8228	0.6487	0.4928	0.3515	0.2769
1.0000	0.9933	0.9177	0.7674	0.6051	0.4622	0.3305	0.2616
0.9750	0.9177	0.8206	0.6824	0.5402	0.4162	0.2998	0.2384
0.8228	0.7674	0.6824	0.5696	0.4549	0.3541	0.2582	0.2060
0.6487	0.6051	0.5402	0.4549	0.3678	0.2897	0.2144	0.1717
0.4928	0.4622	0.4162	0.3541	0.2897	0.2307	0.1733	0.1394
0.3515	0.3305	0.2998	0.2582	0.2144	0.1733	0.1326	0.1073
0.2769	0.2616	0.2384	0.2060	0.1717	0.1394	0.1073	0.0872

(d) $\alpha = 7$; $f_{max} = 15.7$ cycles/degree

Fig. 7 Two-dimensional HVS weighting matrices for the LOT, assuming 256 pels in a line and blocks of 8×8 pels. The relative distance α and maximum frequency f_{max} are indicated.

0.8945	1.0000	0.9209	0.7295	0.5375	0.3865	0.2539	0.2005
1.0000	0.9644	0.8474	0.6684	0.4942	0.3581	0.2362	0.1872
0.9209	0.8474	0.7270	0.5746	0.4289	0.3143	0.2098	0.1663
0.7295	0.6684	0.5746	0.4589	0.3477	0.2581	0.1754	0.1387
0.5375	0.4942	0.4289	0.3477	0.2683	0.2022	0.1404	0.1108
0.3865	0.3581	0.3143	0.2581	0.2022	0.1543	0.1092	0.0862
0.2539	0.2362	0.2098	0.1754	0.1404	0.1092	0.0792	0.0627
0.2005	0.1872	0.1663	0.1387	0.1108	0.0862	0.0627	0.0499

(a) $\alpha = 4$; $f_{max} = 18$ cycles/degree

0.9608	1.0000	0.8236	0.5781	0.3739	0.2485	0.1360	0.1222
1.0000	0.9107	0.7255	0.5121	0.3343	0.2242	0.1239	0.1104
0.8236	0.7255	0.5746	0.4122	0.2747	0.1863	0.1057	0.0915
0.5781	0.5121	0.4122	0.3025	0.2071	0.1423	0.0835	0.0698
0.3739	0.3343	0.2747	0.2071	0.1460	0.1021	0.0622	0.0504
0.2485	0.2242	0.1863	0.1423	0.1021	0.0723	0.0452	0.0361
0.1360	0.1239	0.1057	0.0835	0.0622	0.0452	0.0295	0.0231
0.1222	0.1104	0.0915	0.0698	0.0504	0.0361	0.0231	0.0185

(b) $\alpha = 5$; $f_{max} = 22.4$ cycles/degree

1.0000	0.9676	0.7115	0.4434	0.2512	0.1646	0.0707	0.0878
0.9676	0.8317	0.6001	0.3796	0.2184	0.1433	0.0631	0.0754
0.7115	0.6001	0.4384	0.2857	0.1699	0.1111	0.0516	0.0568
0.4434	0.3796	0.2857	0.1928	0.1191	0.0779	0.0385	0.0387
0.2512	0.2184	0.1699	0.1191	0.0767	0.0507	0.0266	0.0245
0.1646	0.1433	0.1111	0.0779	0.0507	0.0338	0.0182	0.0164
0.0707	0.0631	0.0516	0.0385	0.0266	0.0182	0.0106	0.0087
0.0878	0.0754	0.0568	0.0387	0.0245	0.0164	0.0087	0.0082

(c) $\alpha = 6$; $f_{max} = 26.8$ cycles/degree

1.0000	0.8965	0.5850	0.3231	0.1591	0.1141	0.0347	0.0692
0.8965	0.7254	0.4715	0.2669	0.1343	0.0943	0.0303	0.0561
0.5850	0.4715	0.3165	0.1870	0.0985	0.0663	0.0236	0.0379
0.3231	0.2669	0.1870	0.1157	0.0641	0.0418	0.0165	0.0227
0.1591	0.1343	0.0985	0.0641	0.0375	0.0241	0.0106	0.0123
0.1141	0.0943	0.0663	0.0418	0.0241	0.0159	0.0069	0.0083
0.0347	0.0303	0.0236	0.0165	0.0106	0.0069	0.0035	0.0033
0.0692	0.0561	0.0379	0.0227	0.0123	0.0083	0.0033	0.0046

(d) $\alpha = 7$; $f_{max} = 31.4$ cycles/degree

Fig. 8 Two-dimensional HVS weighting matrices for the LOT, assuming 512 pels in a line and blocks of 8×8 pels. The relative distance α and maximum frequency f_{max} are indicated.

shows weighting matrices containing normalized ζ_{ij} for f_{max} as 9.0, 11.2, 13.4, and 15.7 cycles/degree. They represent $\alpha = 4$, 5, 6, 7, respectively, for $N = 256$. Values for α of 6 or 7 are more representative for broadcast TV viewing. Values of 4 or 5 fit modern PIT needs very well and approximate the situation in which a 256×256 pel image is displayed on the 640×480 resolution mode on a regular home PC monitor, with the observer in front of it, working on the computer. The same procedure is repeated for the matrices in Fig. 8, assuming $N = 512$. For this value of N and the same values of α, the maximum frequencies are 18.0, 22.4, 26.8, and 31.4 cycles per degree.

4 Implementation and Results

A 256×256 pel monochrome image is divided into 8×8 nonoverlapping blocks ($M = 8$) and the LOT is applied to each block. Based on the ac energies, the 8×8 blocks are grouped into eight different equally populated classes ($N_C = 8$). Thus, there are 32×32 blocks in the image ($N_B = 1024$). The dc coefficients are quantized with a uniform 7-bit quantizer, and B_{max} is set to 7. Therefore, the overhead in Eq. (7) is, as previously computed, 4552 bits and the amount of bits needed to code the dc coefficients is 7168. This yields a total of 11,720 bits sent prior to the transmission of the ac coefficients (approximately 0.18 bits/pel). The block classification map for the 256×256 monochrome "Lena" image is shown in Fig. 9. Classes 1 through 8 represent increasing energies of 2-D LOT blocks. Figure 10 shows maps with standard deviations. Classes 1, 3, 6, and 8 are chosen as examples, and the dc coefficient is not computed. The resulting bit-allocation map for the eight classes is presented in Fig. 11. Using these maps and the weighting matrix of Fig. 7 (for $\alpha = 6$), by means of Eq. (9) we get the order for the transmission of the ac coefficients as shown in Fig. 12.

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

Fig. 9 Equally populated 32×32 classification map for the "Lena" image. Classes 1 through 8 represent increasing energies of 8×8 LOT blocks.

Class 1

0.0	8.1	3.9	4.0	2.6	2.9	2.2	2.3
7.7	4.5	3.3	3.0	2.1	2.3	2.1	2.0
3.9	2.6	2.6	2.4	2.1	1.8	1.7	1.8
2.7	2.1	2.3	2.0	2.1	1.7	1.8	1.8
2.1	2.0	1.7	1.7	1.7	1.6	1.7	1.6
2.1	2.0	1.8	1.7	1.9	1.7	1.5	1.6
1.8	1.8	1.7	1.5	1.6	1.6	1.4	1.6
1.9	1.8	1.7	1.6	1.6	1.7	1.5	1.7

Class 3

0.0	33.0	13.6	14.6	5.6	7.6	4.7	3.8
23.3	16.3	11.3	8.3	4.9	3.8	3.5	2.8
10.3	10.5	9.9	5.6	3.9	4.0	3.2	3.4
8.7	8.7	7.6	5.1	3.5	3.3	3.2	2.6
4.0	5.3	5.6	5.2	3.5	3.2	3.2	3.0
4.8	4.5	4.1	4.5	4.3	3.3	3.0	2.4
2.9	2.9	3.4	3.5	3.5	3.6	3.5	3.1
2.8	2.5	2.9	3.4	3.7	4.1	3.7	2.9

Class 6

0.0	121.1	54.8	39.1	22.9	24.1	13.8	12.8
67.3	51.8	42.1	27.4	20.4	17.8	14.0	12.3
29.7	31.1	29.8	25.8	21.4	15.7	15.0	12.1
22.5	20.7	22.1	19.7	16.8	11.3	10.0	7.8
12.9	13.9	12.3	14.0	11.4	12.4	9.8	8.2
12.1	11.4	11.6	8.2	9.4	9.6	6.5	6.6
5.6	5.8	5.8	6.2	5.8	6.2	5.1	5.9
6.2	5.7	5.5	5.3	6.4	5.1	4.4	4.9

Class 8

0.0	316.5	116.1	68.0	41.6	34.0	20.1	20.5
160.7	149.2	81.6	46.3	39.1	32.5	18.8	19.2
47.9	62.2	54.9	45.7	26.6	22.7	18.9	16.8
31.7	31.3	34.3	27.3	27.0	22.6	13.4	13.1
18.1	18.5	21.7	21.3	18.1	15.8	14.2	11.6
13.5	16.8	14.9	11.8	15.5	11.4	9.2	9.4
8.5	10.2	8.4	9.1	8.4	8.9	7.1	7.6
9.1	11.2	9.0	7.5	8.3	7.2	6.2	6.8

Fig. 10 Map with standard deviations of LOT coefficients in each class. Classes 1, 3, 6, and 8 are chosen as examples. The standard deviation for the dc coefficient is not shown.

The ac coefficients are well modeled by a Laplacian probability density function (pdf), but the blocks are classified according to their ac activities. If u is the amplitude of an ac coefficient, the actual important function is no longer its density function $p_U(u)$, but one conditional to the estimated standard deviation $p_U(u|\hat{\sigma})$. If there is just one class ($N_C = 1$), the Laplacian model fits well. At the other extreme, suppose there are as many blocks as classes (the overhead would be enormous). The variances would be computed from one element and would determine its amplitude completely. Therefore, the density would be an impulse. In this extreme case, all quantizers should only have two levels to indicate the sign of the coefficient. As long as we have few classes, these extreme cases do not apply. However, the lowest frequency ac coefficients (which have great influence in the classification process because they are larger) are well apart from having a Laplacian conditional density. As an example for a particular coefficient, suppose its standard deviation is estimated to be very large. This indicates that the coefficients on that coordinate $(m,n) \in \Psi$ belonging to the same class are expected to have high amplitudes, not amplitudes close to zero as in the Laplacian model. Generally, these large coefficients have low frequencies and have large numbers of bits allocated. Coefficients with one or two bits allocated generally do not have a great influence on the ac energy and are very close to the Laplacian model. In our constant distortion rule for bit allocation, we assumed that all the quantizers were optimized using the same pdf. Therefore, we have chosen the Gaussian density as the density model for our Lloyd-Max quantizers due to its greater robustness against pdf mismatches. Tests carried out (for 8 and 16 classes) using two sets of quantizers (for Laplacian and Gaussian pdfs), showed better performance for the Gaussian set of quantizers.

The reestimated standard deviations assume an integer number of bits allocated to each coefficient; hence, if we assume that all quantizer levels may be used, the quantizer should be a midrise one. For one- and two-bit quantizers optimized for a Gaussian input pdf, the inner reconstruction levels (positive or negative) are $0.798\sigma_1$ and $0.453\sigma_2$, respectively, where $\sigma_1 = c$ and $\sigma_2 = 2c$ represent the estimated standard deviations for those coefficients that have been allocated 1 and 2 bits, respectively. It is possible that some null or insignificant coefficients would have to be quantized using relatively high standard deviation values, and must be reconstructed as a nonzero component with a magnitude comparable to the standard deviation. In these cases, nonexistent frequency components emerge, resulting in annoying effects. For this reason, we decided to apply midtread quantizers with three levels and variable length coding, instead of quantizing with two or four levels. The standard

Class 1

7	1	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
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Class 2

7	2	1	1	0	0	0	0
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	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

Class 3

7	3	1	2	0	1	0	0
2	2	1	1	0	0	0	0
1	1	1	0	0	0	0	0
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	0	0	0	0
0	0	0	0	0	0	0	0

Class 4

7	3	2	2	1	1	0	0
2	3	2	1	1	0	0	0
1	1	2	1	1	0	0	0
1	1	1	1	0	0	0	0
0	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	0

Class 5

7	4	3	3	2	2	1	1
3	3	2	2	1	1	0	0
2	2	2	1	1	1	0	0
2	2	1	1	1	0	0	0
1	1	1	1	1	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

Class 6

7	5	3	3	2	2	1	1
4	3	3	2	2	2	1	1
3	3	3	2	2	2	2	1
2	2	2	2	2	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Class 7

7	5	4	3	3	2	2	2
4	4	4	3	3	2	2	2
3	3	3	3	2	2	1	1
2	2	2	2	2	2	2	1
1	2	2	2	2	1	1	1
1	2	1	1	1	1	1	1
0	1	1	1	1	0	0	0
1	1	1	1	0	0	0	0

Class 8

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

Fig. 11 Bit-allocation maps for each class for the "Lena" image. The rate is 1 bit/pel including overhead. This is also the final stage of the PIT.

Class 1

–	137	–	–	–	–	–	–
138	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–

Class 2

–	70	114	119	–	–	–	–
139	124	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–

Class 3

–	40	115	56	–	161	–	–
71	61	111	125	–	–	–	–
116	112	118	–	–	–	–	–
120	126	131	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–

Class 4

–	41	52	57	140	162	–	–
72	31	49	127	144	–	–	–
117	113	54	132	155	–	–	–
121	128	133	151	–	–	–	–
–	145	156	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–

Class 5

–	13	19	27	73	87	181	198
42	32	50	62	146	170	–	–
53	51	55	134	157	172	–	–
58	63	135	152	167	–	–	–
141	147	158	168	175	–	–	–
163	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–

Class 6

–	3	20	28	74	88	182	199
14	33	16	64	76	95	184	202
21	17	24	67	82	99	122	213
59	65	68	79	90	177	194	217
142	148	159	169	176	190	208	225
164	171	173	178	191	205	–	–
–	–	–	–	–	–	–	–
–	–	–	–	–	–	–	–

Class 7

–	4	9	29	43	89	107	149
15	12	6	34	45	96	109	153
22	18	25	37	83	100	187	214
60	66	69	80	91	105	136	218
143	77	84	92	103	192	209	226
165	97	174	179	193	206	222	229
–	185	188	195	210	–	–	–
200	203	215	219	–	–	–	–

Class 8

–	1	10	11	44	47	108	150
5	2	7	35	46	48	110	154
23	8	26	38	85	101	123	160
30	36	39	81	93	106	196	220
75	78	86	94	104	129	211	227
166	98	102	180	130	207	223	230
183	186	189	197	212	224	–	231
201	204	216	221	228	–	–	–

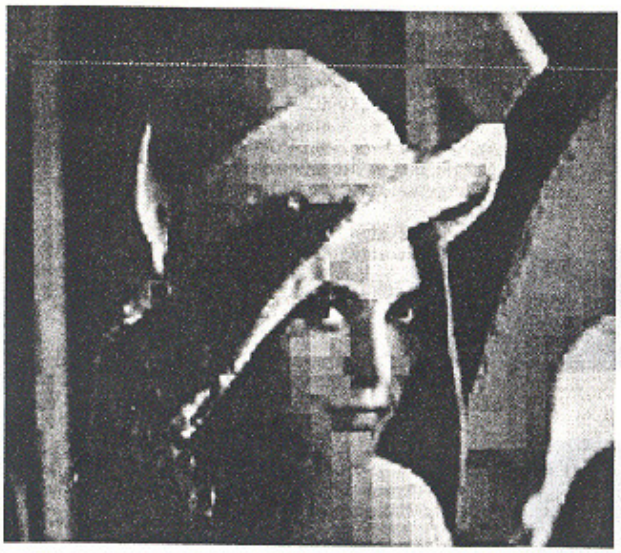
Fig. 12 Transmission order of the LOT coefficients among all classes for the "Lena" image. This order is found using the bit-allocation maps in Fig. 11, weighted by the HVS matrix in Fig. 7(c), according to Eq. (9). The transmission priority rules were defined in Sec. 2 based on these weighted matrices.

Fig. 13 Partially reconstructed images: (a) DCT 0.2 bit/pel, (b) LOT 0.2 bit/pel, (c) DCT 0.3 bit/pel, and (d) LOT 0.3 bit/pel.

deviations for quantization and reconstruction of these coefficients would remain the same, but the distortion rule and the average bit rate would be affected. However, the distortion increase, a result of going from four to three levels in the 2-bit quantizer, is partially compensated by the distortion decrease in going from two to three levels for the 1-bit quantizer. The same occurs with the bit-rate changes. In our simulations, both schemes yielded roughly the same bit rates, with the three-level scheme leading to images with higher signal-to-noise ratios (SNRs).

The HVS-weighted PIT described previously is extended to the 2-D DCT. The weighting matrix was found using the method described in Ref. 2 for $f_{max} = 13.4$ ($\alpha = 6$). Reconstructed images based on both LOT and DCT for several stages are shown in Fig. 13. Critical observation of these images indicates the improved fidelity and absence of block structure during the initial stages when LOT is used. In Table 1, a comparison of both methods is carried out, evaluating the SNR of reconstructed images at several stages for the "Lena" and "Girl" images. Since the HVS weighting is used only for prioritizing the transmission of coefficients, the SNR measure did not incorporate subjective weighting factors. If $u(m,n)$ and $\hat{u}(m,n)$ represent the original and reconstructed image, then the SNR is given by

$$\mathrm{SNR} = 10 \log_{10} \left\{ \frac{\sum_{m=0}^{N-1} \sum_{n=0}^{N-1} u^2(m,n)}{\sum_{m=0}^{N-1} \sum_{n=0}^{N-1} [u(m,n) - \hat{u}(m,n)]^2} \right\} .$$



(e)

(f)

(g)

(h)

Fig. 13 (continued) Partially reconstructed images: (e) DCT 0.4 bit/pel, (f) LOT 0.4 bit/pel, (g) DCT 1.0 bit/pel, and (h) LOT 1.0 bit/pel.

5 Conclusions

A PIT scheme that incorporates adaptive classification in the transform domain and bit allocation based on the rate-distortion theory is presented. A general technique for developing HVS weighting of the transform coefficients is developed. Based on this, HVS weighting matrices applicable to LOT are obtained. The order in which the transform coefficients are transmitted is based on the estimated variances of these coefficients weighted by the human visual system sensitivity, measured in the 2-D LOT domain. Because these variances can be estimated at the receiver, overhead is limited to bit-allocation maps of the classes to which the blocks are grouped and to the classification of the blocks. The transform coefficients for all the classes during each stage are transmitted progressively such that a specified bit rate is reached for each stage. Visual comparison of the reconstructed images based on the LOT and DCT shows that the former yields subjectively superior images compared to the DCT in all stages.

Table 1 SNR (in decibels) resulting from intermediary reconstructed images at several bit rates for the "Lena" and "Girl" images.

Rate(bpp)	SNR			
	LOT	DCT	LOT	DCT
	LENA		GIRL	
0.2	16.10	15.18	17.22	16.21
0.3	19.43	18.41	20.27	19.41
0.4	21.00	20.55	22.68	21.93
0.5	22.74	22.39	24.30	23.79
0.6	23.68	23.29	25.21	24.98
0.8	25.35	25.15	26.98	26.76
1.0	26.81	26.67	28.50	28.27

Acknowledgment

This work was supported in part by Conselho Nacional de Desenvolvimento Cientifico e Tecnologico, CNPq, Brazil, under grant 200.804/90-1.

References

1. K. H. Tzou, "Progressive image transmission: a review and comparison of techniques," *Opt. Eng.* **26,** 581–589 (July 1987).
2. B. Chitprasert and K. R. Rao, "Human visual weighted progressive image transmission," *IEEE Trans. Commun.*, **COM-38,** 1040–1044 (July 1990).
3. K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages and Applications*, Academic Press, San Diego (1990).
4. M. Rabbani and P. W. Jones, *Digital Image Compression Techniques*, SPIE Optical Engineering Press, Bellingham, WA (1991).
5. K. H. Tzou and S. E. Elnahas, "An optimal progressive transmission and reconstruction scheme for transformed images," ICC 86, pp. 413–418, Toronto, Canada (June 1986).
6. W. Gong, K. R. Rao, and M. T. Manry, "Progressive image transmission using Kohonen self organizing feature map," *IEEE Trans. Circuits Sys. Video Technol.* (under review).
7. E. R. Hamilton, "The JPEG standard for still picture coding," Society for Information Display (SID), 1991 International Symposium, Anaheim, CA, May 6–10, 1991.
8. A. Leger, T. Omachi, and E. K. Wallace, "JPEG still picture compression algorithm," *Opt. Eng.* **30,** 947–954 (July 1991).
9. W. Gong, K. R. Rao, and M. T. Manry, "Progressive image transmission using a self-supervised back-propagation neural network," *J. Electronic Imaging*, **1,** 88–94 (Jan. 1992).
10. S. E. Elnahas et al., "Progressive transmission of digital diagnostic images," *Appl. Digital Image Processing VIII*, Proc. SPIE **575,** 48–55 (Aug. 1985).
11. W. H. Chen and W. K. Pratt, "Scene adaptive coder," *IEEE Trans. Commun.* **COM-32,** pp. 225–232 (March 1984).
12. W. H. Chen and C. H. Smith, "Adaptive coding of monochrome and color images," *IEEE Trans. Commun.*, **COM-25,** 1285–1292 (Nov. 1977).
13. K. N. Ngan, "Adaptive transform coding of video signals," *IEE Proc.*, **129,** Pt. F, pp. 28–40 (Feb. 1982).
14. T. Saito, H. Takeo, K. Aizawa, H. Harashima, and H. Miyakawa, "Adaptive discrete cosine transform image coding using gain/shape vector quantization," *Proc. ICASSP '86*, pp. 129–132 (April 1986).
15. Y. S. Ho and A. Gersho, "Classified transform coding of images using vector quantization," *Proc. ICASSP '89*, 1890–1893 (April 1989).
16. J. Y. Nam and K. R. Rao, "Image coding using a classified DCT/VQ based on two channel conjugate vector quantization," *IEEE Trans. Circuits Sys. Video Technol.* **1,** 327–336 (Dec. 1991).
17. K. H. Tzou, T. R. Hsing, and J. G. Dunham, "Applications of physiological human visual system model to image compression," *Proc. SPIE* **504,** 419–424 (1984).
18. S. Ericsson, "Frequency weighted interframe hybrid coding," Rep. TRITA-TTT-8401, Telecommun. Theory, Royal Institute of Technology, Stockholm, Sweden (Jan. 1984).
19. H. Lohscheller, "A subjectively adapted image communication system," *IEEE Trans. Commun.* **COM-32,** 1316–1322 (Dec. 1984).
20. N. B. Nill, "A visual model weighted cosine transform for image compression and quality assessment," *IEEE Trans. Commun.* **COM-33,** 551–557 (June 1985).
21. K. N. Ngan, K. S. Leong, and H. Singh, "Cosine transform coding incorporating human visual system model," *Proc. SPIE* **707,** 165–171 (Sep. 1986).
22. P. M. Cassereau, D. H. Staelin, and G. De Jager, "Encoding of images based on a lapped orthogonal transform," *IEEE Trans. Commun.* **COM-37,** 189–193 (Feb. 1989).
23. H. S. Malvar and D. H. Staelin, "The LOT: Transform coding without blocking effects," *IEEE Trans. Acoust., Speech, Signal Proc.* **ASSP-37,** 553–559 (April 1989).
24. H. S. Malvar, "Reduction of blocking effects in image coding with a lapped orthogonal transform," *Proc. ICASSP '88*, 781–784 (April 1988).
25. H. S. Malvar, "Lapped transforms for efficient transform/subband coding," *IEEE Trans. Acoust., Speech, Signal Proc.* **ASSP-38,** 969–978 (June 1990).
26. H. S. Malvar, *Signal Processing with Lapped Transforms*, Artech House, Norwood, MA (1992).
27. A. N. Akansu and F. E. Wadas, "On lapped orthogonal transforms," *IEEE Trans. Signal Proc.* **40,** 439–443 (Feb. 1992).
28. W. E. Lynch and A. R. Reibman, "The lapped orthogonal transform for motion-compensated video compression," *Proc. SPIE* **1605,** 285–296 (1992).
29. H. Jozawa and H. Watanabe, "Compatible coding by lapped orthogonal transform," presented at ITEC 90, 1990 ITE Annual Convention.
30. H. Jozawa and H. Watanabe, "Intrafield/Interfield adaptive lapped transform for compatible HDTV coding," presented at 4th International Workshop on HDTV and Beyond, Torino, Italy, Sep. 4–6, 1991.
31. S. Venkatraman, J. Y. Nam, and K. R. Rao, "Classified transform vector quantization of images using the lapped orthogonal transform," *Proc. SPIE* (in press).
32. M. Vetterli and D. Le Gall, "Perfect reconstruction filter banks: some properties and factorizations," *IEEE Trans. Acoust., Speech, Signal Proc.* **ASSP-37,** 1057–1071 (July 1989).
33. R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ (1983).

Ricardo L. de Queiroz received the BSc degree from Universidade de Brasilia in 1987 and the MSc degree from Universidade Estadual de Campinas, Brazil, in 1990, both in electrical engineering. In 1990–1991, he was with the DSP research group at Universidade de Brasilia as a research associate. He is currently enrolled in the PhD program at University of Texas at Arlington. His main interests are multirate signal processing, filter banks, image and audio compression, and image databases. He is a member of IEEE and the Brazilian Telecommunications Society.

K. R. Rao received his BE degree from the University of Madras in 1952, MSEE and MSNE degrees from the University of Florida, Gainesville, in 1959 and 1960, respectively, and the PhD degree in electrical engineering from the University of New Mexico, Albuquerque, in 1966. Since 1966, he has been with the University of Texas at Arlington (UTA) where he is currently a professor of electrical engineering. He has published extensively in reviewed technical journals in the areas of discrete orthogonal transforms and digital image coding. He, along with two other researchers, introduced the discrete cosine transform in 1975, which has since become very popular in digital signal processing. He has organized and conducted short courses and conferences on thermoelectric energy conversion from 1969–1992. He is the coauthor of the books *Orthogonal Transforms for Digital Signal Processing.* (Springer-Verlag, 1975), *Fast Transforms: Analyses and Applications* (Academic Press, 1982), and *Discrete Cosine Transform—Algorithms, Advantages, and Applications* (Academic Press, 1990), as well as coauthor or editor on other works.