AN EFFICIENT WEIGHT OPTIMIZATION ALGORITHM FOR IMAGE REPRESENTATION USING NONORTHOGONAL BASIS VECTORS

Yuk-Hee Chan and Wan-Chi Siu

Department of Electronic Engineering
The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong

ABSTRACT

Though image-coding techniques that employ subsets of nonorthogonal basis images chosen from two or more transform domains have been shown consistently to yield higher image quality than those based on one transform for a fixed compression ratio, they have not been widely employed due to their very high computational complicity of existing realization approaches. This paper presents a new realization approach for mixed-transform image representation. Computational complexity can be greatly reduced compared with existing approaches.

1. INTRODUCTION

Recently, mixed-transform coding techniques have been proposed to represent a signal from subsets of basis vectors selected from multiple transform domain[1-4]. These coders have been shown to yield higher image quality compared with those based on a single transform. However, they have not been widely employed because of their very high computational complexity. This paper presents a new realization approach for mixed-transform image representation. The approach greatly reduces the computational complexity compared with existing approaches.

2. ALGORITHM

In block-based transform image coding, a block of image can be generally represented as $x=Tc$, where x and c are respectively the lexicographically ordered original image and its corresponding transform coefficient vector, and T is the corresponding 2D transformation operator. Note that the ith column of T is actually the lexicographically ordered basis vector associated with the ith transform coefficient. Obviously, if different unitary transforms, say T_a and T_b, are exploited, their

corresponding coefficient vectors will be different and we have $x = T_a c_a = T_b c_b$.

Consider a mixed-transform image coder which selects dominant components in the transform domains of these two transforms. The image x is encoded with n_a basis vectors of T_a and n_b basis vectors of T_b. Then the reconstructed version of x, say $\hat{x}$, can be represented as

$$\hat{x} = T_a u + T_b v \qquad (1)$$

where u and v are lexicographically ordered vectors whose elements equal to the weights of selected T_a and T_b basis vectors respectively and are zero otherwise. In the transform domain of T_a, it can be represented as

$$T_a^{-1}\hat{x} = u + T_a^{-1}T_b v \qquad (2)$$

In non-matrix form, we have

$$\hat{c}_{a,i} = u_i + \sum_{j \in \Lambda_b} \alpha_{ij} v_j \qquad for \ i \in \Lambda_a$$

$$\hat{c}_{a,i} = \sum_{j \in \Lambda_b} \alpha_{ij} v_j \qquad for \ i \notin \Lambda_a \qquad (3)$$

where $\hat{c}_{a,i}$, u_i, v_i are the ith elements of $T_a^{-1}\hat{x}$, u and v respectively, and, α_{ij} is the ith element of the jth column of $T_a^{-1}T_b$. Here, Λ_a is the set of indices whose associated basis vectors of T_a are selected and Λ_b is the set of indices whose associated basis vectors of T_b are selected.

Our proposed approach finds the best representation of a given image by repeatedly carrying out the following 2 steps until the reduction of the distortion is less than a predefined threshold or a particular criterion is achieved: (i) select a new basis vector via a marginal analysis and update Λ_a and Λ_b accordingly, and (ii) determine the weights of the selected basis vectors for the updated Λ_a and Λ_b. The initial condition

is $\Lambda_a = \Lambda_b = \{\ \}$. In the following, we will describe the 2 steps in detail.

A. Determine the weights of selected basis vectors:

If minimum distortion criterion is exploited, the weights of the selected basis vectors should be determined in a way that the reconstructed $\hat{x}$ minimizes the distortion $d = ||\mathbf{x} - \hat{x}||^2$. As $\mathbf{T_a}$ and $\mathbf{T_b}$ are unitary, we have

$$d = ||\mathbf{T_a^{-1}}(\mathbf{x} - \hat{x})||^2 = \sum_{i=0}^{N-1} (c_{a,i} - \hat{c}_{a,i})^2 \qquad (4)$$

where N is the dimension of $\mathbf{x}$ and $c_{a,i}$ is the ith element of $\mathbf{c_a}$. The minimum distortion can be achieved by separately minimizing each component in the transform domain of $\mathbf{T_a}$.

Obviously, for given Λ_a and Λ_b, we can always make $\hat{c}_{a,i} = c_{a,i}$ to reduce the distortion of this component to zero by adjusting u_i if $i \in \Lambda_a$. Hence, in order to minimize the distortion d, all we need to do is to minimize

$$E = \sum_{i \notin \Lambda_a} (c_{a,i} - \hat{c}_{a,i})^2 = \sum_{i \notin \Lambda_a} \left(c_{a,i} - \sum_{j \in \Lambda_b} \alpha_{ij} v_j\right)^2 \qquad (5)$$

To simplify the formulation of the algorithm, we can rewrite eqn. (5) in matrix form as

$$E = ||\vec{c} - \mathbf{A}\vec{v}||^2 \qquad (6)$$

where $\vec{c}$ and $\vec{v}$ are the vectors whose elements are $c_{a,i}$'s sorted according to $i \notin \Lambda_a$ and v_i's sorted according to $i \in \Lambda_b$ respectively, and $\mathbf{A}$ is the corresponding matrix composed of α_{ij}'s. By minimizing E with respect to $\vec{v}$, we have

$$\vec{v} = (\mathbf{A^t A})^{-1}\mathbf{A^t}\vec{c} \qquad (7)$$

Note that $\mathbf{A}$ is a submatrix of $\mathbf{T_a^{-1} T_b}$. As both $\mathbf{T_a}$ and $\mathbf{T_b}$ are orthogonal transform kernels, matrix $\mathbf{A^t A}$ is of full rank and its inverse exists. Also note the matrix $\mathbf{A^t A}$ is of size $n_b \times n_b$. One can always make $n_a \geq n_b$ as it is free to pick any particular transform as $\mathbf{T_a}$. That implies $n_b \leq n/2$, where $n = n_a + n_b < N$ is the total basis vectors currently selected. In typical block-based transform coding, we usually have N=64. Besides, our approach starts with small n. That makes the size of $\mathbf{A^t A}$ small and hence the direct computation of $(\mathbf{A^t A})^{-1}$ practical.

After obtaining v_j's, the weights of the selected $\mathbf{T_a}$ basis vectors can be determined with $u_i = c_{a,i} - \sum_{j \in \Lambda_b} \alpha_{ij} v_j$ for $i \in \Lambda_a$.

B. Select a new basis vector:

Suppose we have obtained the optimal set of u_i's and v_i's for a given combination of Λ_a and Λ_b and want to include one more basis vector. Without loss of generality, we assume the newly included basis vector comes from the transform domain of $\mathbf{T_a}$ and its index is m. Then, based on eqns. (3) and the analysis we have made, the guaranteed reduction in distortion is given as

$$\Delta E_{a,m} = \left(c_{a,m} - \sum_{j \in \Lambda_b} \alpha_{mj} v_j\right)^2 \qquad (8)$$

One can determine the best candidate among those not yet selected basis vectors of $\mathbf{T_a}$ by selecting the m whose $\Delta E_{a,m}$ is maximum. In order to reduce the complexity, one can just search the m which maximizes

$$\Delta E'_{a,m} = \left|c_{a,m} - \sum_{j \in \Lambda_b} \alpha_{mj} v_j\right| \qquad (9)$$

as the same candidate will be picked. By making use of the same approach, we have the best candidate among those not yet selected basis vectors of $\mathbf{T_b}$. By comparing their guaranteed reductions in distortion, one can determine which one should be added.

3. COMPUTATIONAL COMPLEXITY

First of all, one has to carry out two transforms to compute $\mathbf{c_a}$ and $\mathbf{c_b}$. Since transforms which have fast realization algorithms are generally exploited, this overhead can be of an order of $N log_2 N$ multiplications.

At a particular stage, assume that we have already had n_a $\mathbf{T_a}$ and n_b $\mathbf{T_b}$ coefficients. Then $n_b(N - n_a) + n_a(N - n_b)$ multiplications and $n_b(N - n_a) + n_a(N - n_b)$ additions are required to search the best basis vector to be added. After this, one has to determine the weights of the selected basis vectors. To compute v_j's, if $(\mathbf{A^t A})^{-1}\mathbf{A^t}$ is precomputed, roughly $n_b(N - n_a)$ multiplications and $n_b(N - n_a)$ additions will be required. The exact number of operations required depends on whether the newly added basis vector is a basis vector of $\mathbf{T_a}$ or $\mathbf{T_b}$. As for computing u_i's, about $n_a n_b$ multiplications and $n_a n_b$ additions are required. In summary, for each iteration, the number of multiplications

required is $n_b(N - n_a) + n_a(N - n_b) + n_b(N - n_a) + n_a n_b = Nn + Nn_b - 2n_a n_b \leq 3nN/2$.

Suppose M is the total number of basis vectors selected at the end. The total computational complexity in terms of number of multiplications is then bounded by $\sum_{n=0}^{M-1} 3nN/2 < 3NM^2/4$. The total number of additions required is also bounded by this figure.

Like the existing mixed-transform coding algorithm [4], the proposed approach iteratively builds up a more accurate representation by including additional basis vectors one at a time. An iterative approach developed with a gradient descendent algorithm is applied in [4] to compute the weights of the selected vectors. It requires approximately $2nN$ multiplications for each iteration and takes time to converge. However, ours for the same function is non-iterative and takes only less than $nN/2$ multiplications totally.

4. SIMULATIONS

The proposed mixed-transform algorithm was verified through simulations with the discrete cosine transform (DCT) and Haar transform(DHT). Similar performance like those reported in [4] were obtained. Figure 1 shows the coding performance of various image coding schemes in coding "Lenna" which is of size 512×512. The parameter η is defined as the total number of transform coefficients retained to reconstruct the output image divided by the image size. It is obvious that the one using mixed transform provides a better Peak-to-peak Signal Noise Ratio (PPSNR) especially when the value of η is high.

Figure 2 shows the simulation results of coding another testing image "Baboon". The original image is of size 512×512. It was respectively transformed to the DCT, DHT and mixed transform domains. Only 20% of the most significant 512×512 transform coefficients were used to reconstruct the output image in each case. Figures 2(b)-(d) respectively show some details of the reconstructed images. They are enlarged for comparison. One can see the superiority of the scheme using mixed transform by examining the details around the eyebrow in the figures.

5. CONCLUSION

In this paper, a simple and efficient mixed-transform coding algorithm is presented. This algorithm provides an excellent coding performance and greatly reduces the computational complexity compared to existing re-
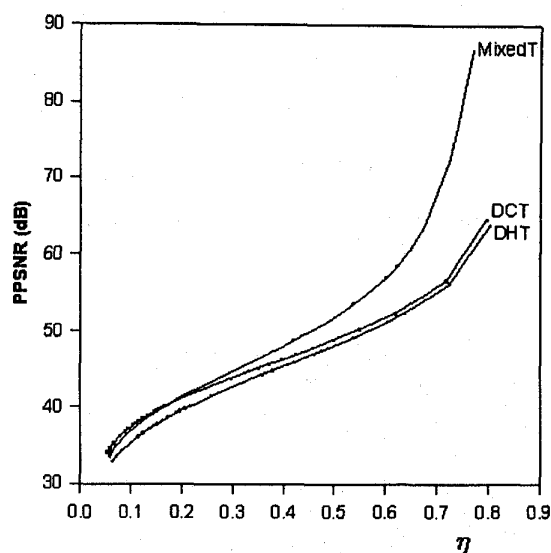
PPSNR vs. No. of Retained coefs


Figure 1: Performance of various coding schemes

alization approaches, which makes mixed-transform image coding practical.

6. REFERENCES

[1] W.B. Mikhael and A. Spanias, "Accurate representation of time varying signals using mixed transforms with applications to speech," IEEE Trans. on Circuits and Systems, Vol.36, No.2, pp.329-331, Feb, 1989.

[2] W. B. Mikhael and A. Ramaswamy, "Application of multitransforms for lossy image representation," IEEE trans. on Circuits and Systems-II, Vol.41, No.6, pp.431-434, Jun 1994.

[3] W. B. Mikhael and A. Ramaswamy, "Resolving images in multiple transform domains with applications," Digital Signal Processing: A review journal, Vol.5, No.2, pp.81-90, Apr 1995.

[4] W. B. Mikhael and A. P. Berg, "Image representation using nonorthogonal basis images with adaptive weight optimization," IEEE Signal Processing Letters, Vol.3, No.6, pp.165-167, Jun 1996.
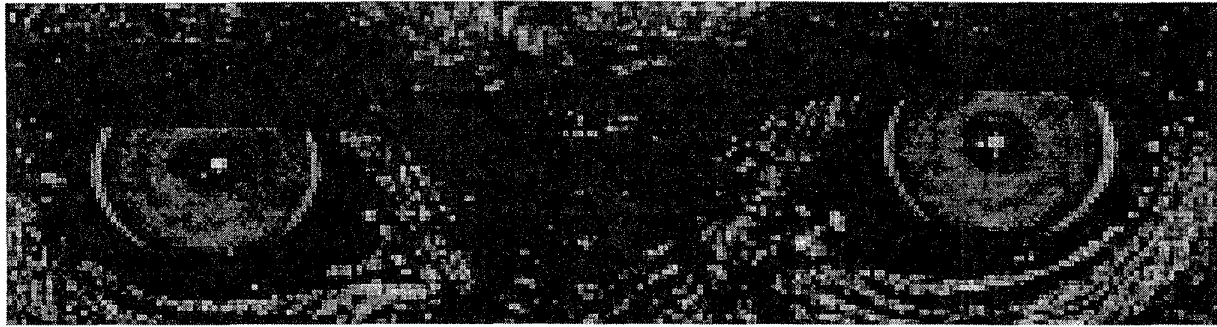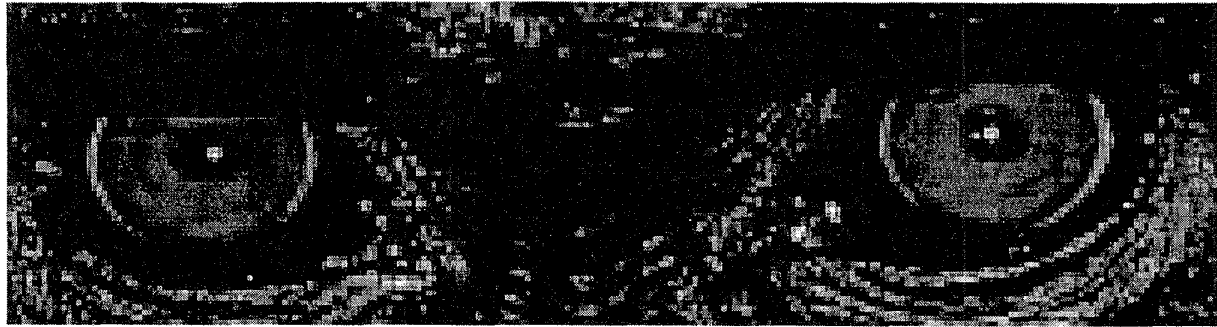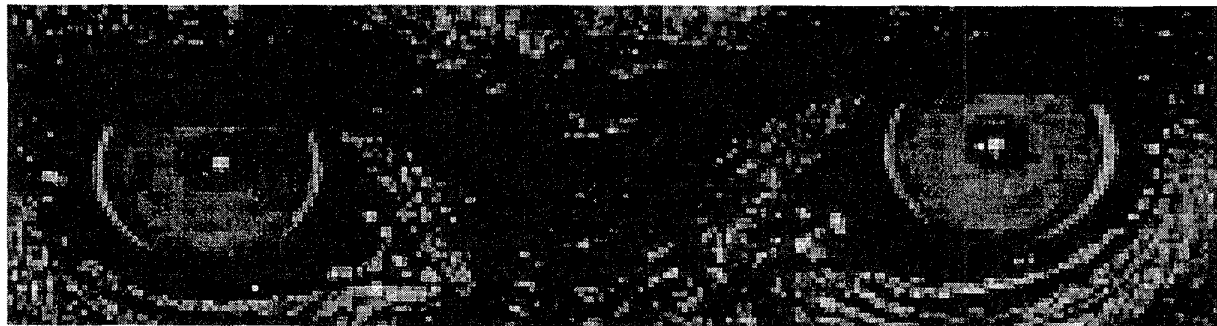
Figure 2: Enlarged "Baboon" and its encoded results (η=0.2): (a). Original (b). DCT-encoded result (c). Mixed-transform-encoded result and (d). DHT-encoded result