An Efficient Weight Optimization Algorithm for Image Representation Using Nonorthogonal Basis Images

Yuk-Hee Chan, *Member, IEEE*

***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 complexity 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.**

***Index Terms*— Image coding, image representation, matching pursuits, nonorthogonal transform.**

I. Introduction

RECENTLY, mixed-transform coding techniques have been proposed to represent a signal from subsets of basis vectors selected from multiple transform domain [1]–[6] as a signal may have some spectral features that cannot be efficiently represented by a particular transform. 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.

II. Algorithm

In block-based transform image coding, a block of image can be generally represented as $\boldsymbol{x} = \boldsymbol{T}\boldsymbol{c}$, where $\boldsymbol{x}$ and $\boldsymbol{c}$ are, respectively, the lexicographically ordered original image and its corresponding transform coefficient vector, and $\boldsymbol{T}$ is the corresponding two-dimensional (2-D) transformation operator. Note that the ith column of $\boldsymbol{T}$ is actually the lexicographically ordered basis vector associated with the ith transform coefficient. Obviously, if different unitary transforms, say $\boldsymbol{T_a}$ and $\boldsymbol{T_b}$, are exploited, their corresponding coefficient vectors will be different and we have $\boldsymbol{x} = \boldsymbol{T_a c_a} = \boldsymbol{T_b c_b}$.

Manuscript received August 12, 1997. This work was supported by The Hong Kong Polytechnic University under Grant A/C 351.504.A3.420. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. R. Ansari.

The author is with the Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Kowloon, Hong Kong (e-mail: enyhchan@polyu.edu.hk).

Publisher Item Identifier S 1070-9908(98)05930-6.

Consider a mixed-transform image coder which selects dominant components in the transform domains of these two transforms. The image $\boldsymbol{x}$ is encoded with n_a basis vectors of $\boldsymbol{T_a}$ and n_b basis vectors of $\boldsymbol{T_b}$. Then the reconstructed version of $\boldsymbol{x}$, say $\hat{\boldsymbol{x}}$, can be represented in the transform domain of $\boldsymbol{T_a}$ as $\boldsymbol{T_a^{-1}}\hat{\boldsymbol{x}} = \boldsymbol{u} + \boldsymbol{T_a^{-1}T_b v}$, where $\boldsymbol{u}$ and $\boldsymbol{v}$ are lexicographically ordered vectors whose elements equal to the weights of selected $\boldsymbol{T_a}$ and $\boldsymbol{T_b}$ basis vectors respectively and are zero otherwise. In nonmatrix form, we have

$$\hat{c}_{a,i} = \begin{cases} u_i + \sum_{j\in\Lambda_b} \alpha_{ij} v_j & \text{for } i \in \Lambda_a \\ \sum_{j\in\Lambda_b} \alpha_{ij} v_j & \text{for } i \notin \Lambda_a \end{cases} \tag{1}$$

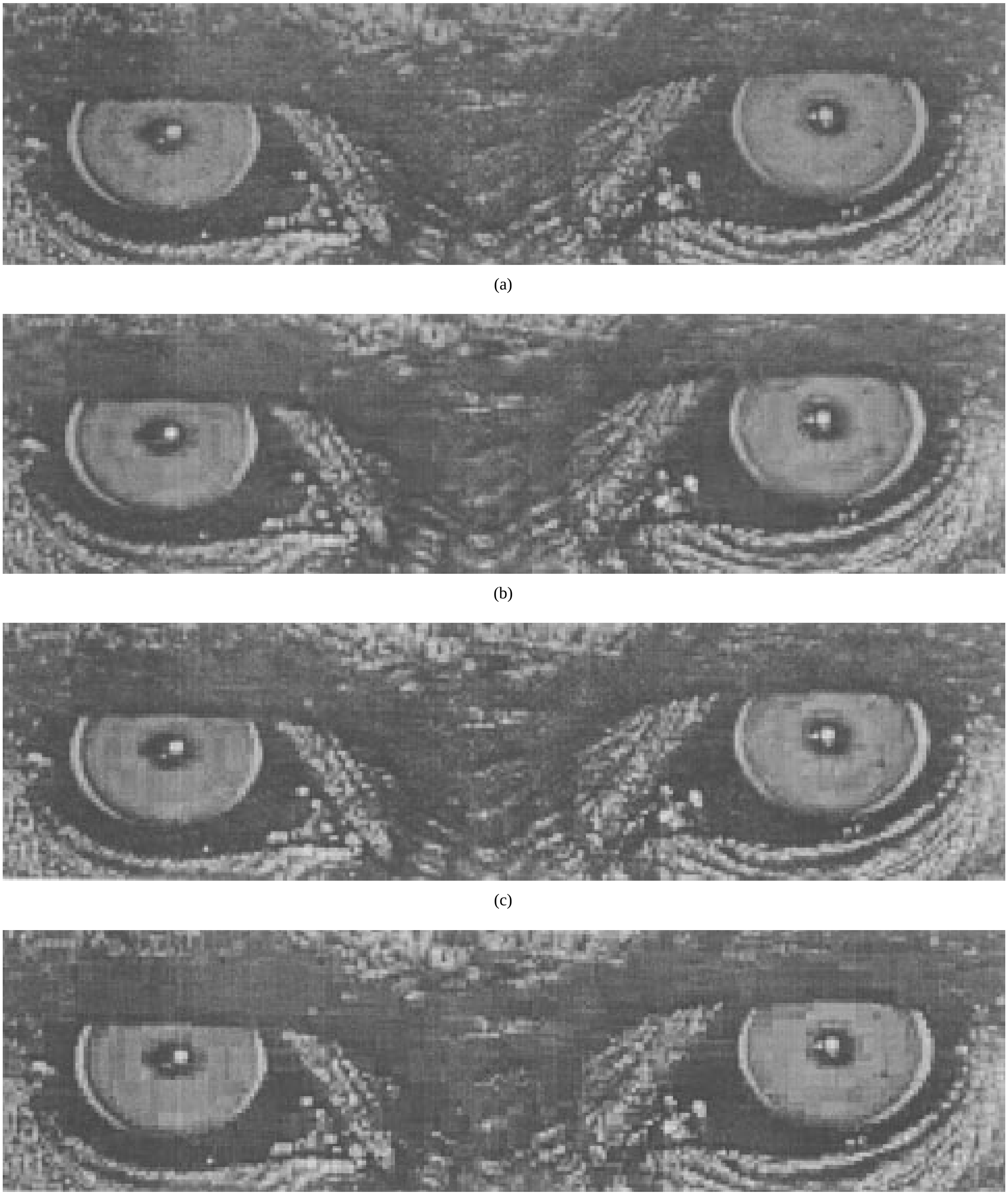
where $\hat{c}_{a,i}, u_i, v_i$ are the ith elements of $\boldsymbol{T_a^{-1}}\hat{\boldsymbol{x}}, \boldsymbol{u}$, and $\boldsymbol{v}$, respectively, and α_{ij} is the ith element of the jth column of $\boldsymbol{T_a^{-1}T_b}$. Here, Λ_a is the set of indices whose associated basis vectors of $\boldsymbol{T_a}$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

Conventional matching pursuit algorithms such as [7] determine the weights one by one iteratively without jointly optimizing them, and hence, the combination of weights found may not provide an optimal representation of a signal. Our proposed approach finds the best representation of a given image by repeatedly carrying out the following two 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) jointly optimize 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 two 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 $\hat{\boldsymbol{x}}$ minimizes the distortion $d = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2$. As $\boldsymbol{T_a}$ and $\boldsymbol{T_b}$ are unitary, we have $d = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2 = \|\boldsymbol{T_a^{-1}}(\boldsymbol{x} - \hat{\boldsymbol{x}})\|^2 = \Sigma_{i=0}^{N-1}\ (c_{a,i} - \hat{c}_{a,i})^2$, where N is the dimension of $\boldsymbol{x}$ and $c_{a,i}$ is the ith element of $\boldsymbol{c_a}$. The minimum distortion can be achieved by separately minimizing each component in the transform domain of $\boldsymbol{T_a}$.

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



(a)

(b)

(c)

(d)

Fig. 1. Enlarged baboon image and its encoded results. (a) Original. (b) DCT-encoded result. (c) Mixed-transform-encoded result. (d) DHT-encoded result.

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

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

To simplify the formulation of the algorithm, we can rewrite (2) in matrix form as $E = ||\vec{\boldsymbol{c}} - \boldsymbol{A}\vec{\boldsymbol{v}}||^2$, where $\vec{\boldsymbol{c}}$ and $\vec{\boldsymbol{v}}$ 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 $\boldsymbol{A}$ is the corresponding matrix composed of α_{ij}'s. By minimizing E with respect to $\vec{\boldsymbol{v}}$, we have $\vec{\boldsymbol{v}} = (\boldsymbol{A}^t\boldsymbol{A})^{-1}\boldsymbol{A}^t\vec{\boldsymbol{c}}$.

Note that $\boldsymbol{A}$ is a submatrix of $\boldsymbol{T_a^{-1}T_b}$. As both $\boldsymbol{T_a}$ and $\boldsymbol{T_b}$ are orthogonal transform kernels, matrix $\boldsymbol{A^tA}$ is of full rank and its inverse exists. Also note the matrix $\boldsymbol{A^tA}$ 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 $\boldsymbol{T_a}$. 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 $\boldsymbol{A^tA}$ small and hence the direct computation of $(\boldsymbol{A^tA})^{-1}$ practical.

After obtaining v_j's, the weights of the selected $\boldsymbol{T_a}$ basis vectors can be determined with $u_i = c_{a,i} - \Sigma_{j\in\Lambda_b}\ \alpha_{ij}v_j$ 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 $\boldsymbol{T_a}$ and its index is m. Then, based on (1) and the analysis we have made, the guaranteed reduction in distortion is given as $\Delta E_{a,m} = (c_{a,m} - \Sigma_{j\in\Lambda_b}\ \alpha_{mj}v_j)^2$. One can determine the best candidate among those not yet selected basis vectors of $\boldsymbol{T_a}$ by selecting the m whose $\Delta E_{a,m}$ is maximum. In order to reduce the complexity, one can just search the m that maximizes $\Delta E'_{a,m} = |c_{a,m} - \Sigma_{j\in\Lambda_b}\ \alpha_{mj}v_j|$ 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 $\boldsymbol{T_b}$. By comparing their guaranteed reductions in distortion, one can determine which one should be added.

III. Computational Complexity

First of all, one has to carry out two transforms to compute $\boldsymbol{c_a}$ and $\boldsymbol{c_b}$. 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 $n_a\boldsymbol{T_a}$ and $n_b\boldsymbol{T_b}$ 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 $(\boldsymbol{A^tA})^{-1}\boldsymbol{A^t}$ 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 $\boldsymbol{T_a}$ or $\boldsymbol{T_b}$. As for computing u_i's, about n_an_b multiplications and n_an_b additions are required. In summary, for each iteration, the number of multiplications required is $n_b(N-n_a)+n_a(N-n_b)+n_b(N-n_a)+n_an_b = Nn + Nn_b - 2n_an_b \leq 3nN/2$.

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 $\Sigma_{n=0}^{M-1}\ 3nN/2 < 3NM^2/4$. 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 noniterative and takes only less than $nN/2$ multiplications totally.

IV. Simulations

The proposed mixed-transform algorithm was verified through simulations with the discrete cosine transform (DCT) and Haar transform (DHT). Fig. 1 shows the simulation results of coding the baboon image. The original image is of size 512 $\times$ 512. It was partitioned into subimages of size 8 $\times$ 8 and then each subimage was, respectively, transformed to the DCT, DHT, and mixed transform domains. Only the most significant 0.2 $\times$ 512 $\times$ 512 transform coefficients were retained to reconstruct the output image in each case. Fig. 1(b)–(d), respectively, shows 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. The PPSNR's of the images encoded with the DCT, mixed-transform and DHT are 29.73, 30.60, and 28.71 dB, respectively.

V. Conclusion

In this letter, 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 realization approaches, which makes mixed-transform image coding practical.

References

[1] W. B. Mikhael and A. Spanias, "Accurate representation of time varying signals using mixed transforms with applications to speech," *IEEE Trans. Circuits Syst.*, vol. 36, pp. 329–331, Feb. 1989.

[2] W. B. Mikhael and A. Ramaswamy, "Application of multitransforms for lossy image representation," *IEEE Trans. Circuits Syst.*, vol. 41, pp. 431–434, June 1994.

[3] ——, "Resolving images in multiple transform domains with applications," *Digital Signal Process.: Rev. J.*, vol. 5, 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 Lett.*, vol. 3, pp. 165–167, June 1996.

[5] R. Neff and A. Zakhor, "Very low bit-rate video coding based on matchimg pursuits," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 158–171, Feb. 1997.

[6] M. R. Banham and J. C. Brailean, "A selective update approach to matching pursuits video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 119–129, Feb. 1997.

[7] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Processing*, vol. 41, pp. 3397–3415, Dec. 1993.