Fast Matching Pursuit Video Coding by Combining Dictionary Approximation and Atom Extraction

Jian-Liang Lin, Wen-Liang Hwang, *Senior Member, IEEE*, and Soo-Chang Pei, *Fellow, IEEE*

***Abstract*—In this paper, we propose a systematic approach that approximates a target dictionary to reduce the complexity of a matching pursuit encoder. We combine calculation of the inner products and maximum atom extraction of a matching pursuit video coding scheme based on eigendictionary approximation and tree-based vector quantization. The approach makes the codec design and optimization cleaner and more systematic than previous dictionary approximation methods. We vary the quality of approximation to demonstrate the tradeoff between computational complexity and coding efficiency. The experiment results show that our codec achieves speed-up factors of up to 100 with a performance loss of less than 0.1 dB. We use double-stimulus impairment scale scores to evaluate the perceptual quality of our approach for different levels of complexity.**

***Index Terms*—Fast algorithm, matching pursuit (MP), tree-based vector quantization (VQ), video coding.**

I. Introduction

EFFICIENTLY encoding motion residuals is essential for low-delay video applications in which videos are encoded by hybrid motion compensation and a residual encoding structure. Matching pursuit (MP), first proposed by Mallat and Zhang in [12], decomposes a signal into a linear combination of bases within an overcomplete dictionary (frame). In [14], Neff and Zakhor show that using MP to code motion residuals performs better than using the discrete cosine transform (DCT) in terms of the peak signal-to-noise ratio (PSNR) and perceptual quality at very low bit rates. The results reported in [10] also demonstrate that the fined-grained scalable (FGS) MP codec performs better than MPEG-4 FGS at very low bit rates. In a transform-based decoder, loop filtering and post processing are usually applied at very low bit rates to remove blocky and ringing artifacts, but an MP decoder can achieve comparable quality without the two processes [13]. Because MP is a data-dependent frame-based representation, an MP codec technique cannot be directly translated from conventional transform-based approaches. Therefore, new MP coding techniques have been developed to deal with quantization noise in the MP algorithm [15], [24], multiple description coding for reliable transmission [21], scalable bitstream generation [10], [1], [23], [19], and dictionary learning and adaptation [4].

Because an MP encoder uses an iterative algorithm, and each iteration performs many inner product calculations, its computational cost is higher than that of transform-based methods. Common approaches that reduce the complexity of an MP encoder are dictionary approximation and suboptimal atom extraction. The latter attempts to find an atom within a local search area. The most popular algorithms of this approach are proposed in [14] and [1], whereby the next atom is found in the block of the largest (weighted) energy. Meanwhile, the results in [11] show that better performance with lower complexity is achievable, provided that the next atom is found in multiple blocks. Dictionary approximation, on the other hand, tries to reduce the complexity of the inner products by using a dictionary with a low computational cost to approximate the target dictionary. The works in [18], [22], [3], and [16] are representative of this approach. We are particulary interested in the two-stage approach in [18] and [16] because it is extremely efficient. The approach approximates a basis by a linear combination of the elementary functions. Thus, by computing the inner products of an MP residual and the elementary functions, the inner products of the MP residual and bases can be obtained. Fig. 1(a) shows the structure of the two-stage dictionary.

An efficient implementation of the two-stage structure is described in [16]. However, the elementary dictionary and the order of bases are heuristically chosen. Developing a systematic approach for selecting the elementary functions to approximate a target dictionary and substantially reducing the complexity while maintaining low performance loss are essential for the success of the approach. The method in [2] uses the orthonormal transform between the elementary functions and the bases. According to principal component analysis (PCA), the optimal elementary functions that approximate a dictionary with an orthonormal transform are the eigenfunctions of the dictionary. This structure does not necessarily yield the most efficient approximation of a target dictionary; however, it is a systematic approach for selecting elementary functions, and we have exploited it by combining it with vector quantization (VQ) to find an atom, as is shown in Fig. 1(b). Because the dictionary is approximated by its eigenfunctions, the vector formed by the coefficients of projecting a basis to the eigenfunctions corresponds to a point in the space spanned by the eigenfunctions. The number of points in the space is $|\mathcal{D}|$, which is the size of the dictionary. The VQ approach can then be applied to partition the space into $|\mathcal{D}|$ components, where the points are the centroids of the components. We can design VQ so that if the coordinate of a block, which is obtained from the inner products of the block

Manuscript received January 10, 2004; revised March 31, 2006. This paper was recommended by Associate Editor O. Al-Shaykh.

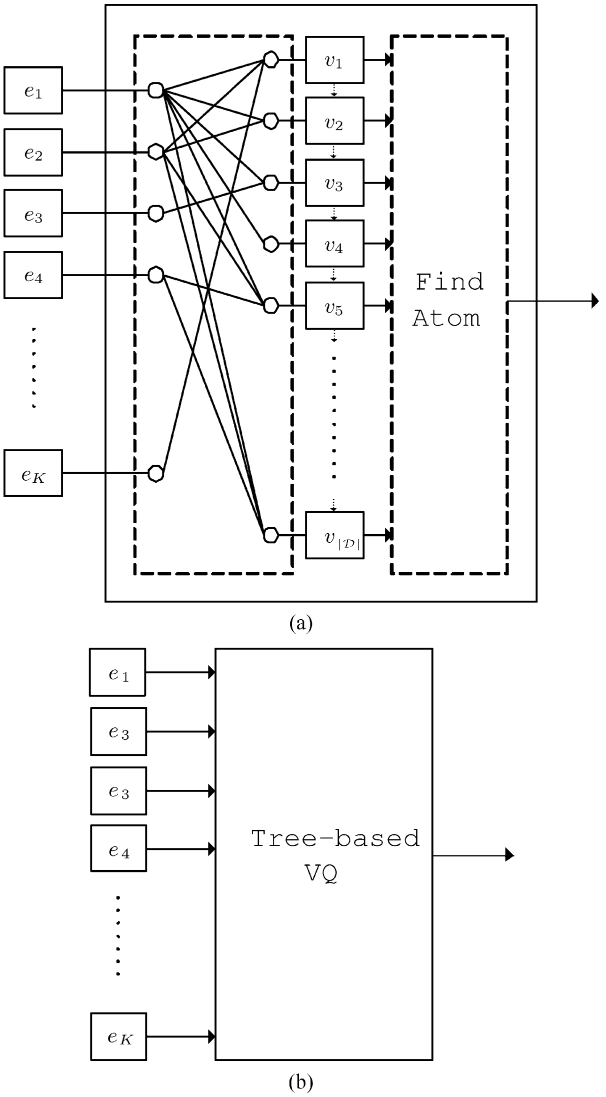
J.-L. Lin was with the Institute of Information Science, Academia Sinica, Taipei 115, Taiwan, R.O.C. He is now with the Institute of Communications Engineering, National Sun Yat-Sen University, Kaohsiung 804, Taiwan, R.O.C. (e-mail: jian@iis.sinica.edu.tw).

W.-L. Hwang is with the Institute of Information Science, Academia Sinica, Taipei 115, Taiwan, R.O.C. (e-mail: whwang@iis.sinica.edu.tw).

S.-C. Pei is with the Department of Electrical Engineering, National Taiwan University, 10617 Taiwan, R.O.C. (e-mail: pei@cc.ee.ntu.edu.tw).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2007.903120



(a)

(b)

Fig. 1. (a) Block diagram of the two-stage dictionary in [18], [16]. The arrows indicate the order of the bases in the approximated dictionary. By ordering the bases, a later basis can be economically represented as a linear combination of its previous bases using only a few elementary functions. (b) The proposed combined design in which the components enclosed in the box in (a) are replaced by a tree-based VQ process.

and the eigenfunctions, lies in a component whose centroid corresponds to a basis, then with the basis the block will have the largest absolute inner product. We then impose tree-based VQ on the centroids to find the basis efficiently.

The elementary functions must be simple so that the inner products with them can be implemented efficiently. However, the eigenfunctions have complex structures, so they should be approximated further. We therefore use a low cost decimated Haar filter bank to approximate the eigenfunctions. The complete process, shown in Fig. 2, is comprised of two stages. The first obtains the inner products with Haar wavelets; and the second obtains the inner products with the eigenfunctions, followed by VQ to find an atom. We call our approach the two-stage VQ approach.

Having introduced the two-stage VQ structure, we present the technical part of our method in Section II. The computational complexity is analyzed and compared to other methods in Section III. The performance of our approach is objectively and subjectively evaluated in Section IV. Section V presents our conclusions.

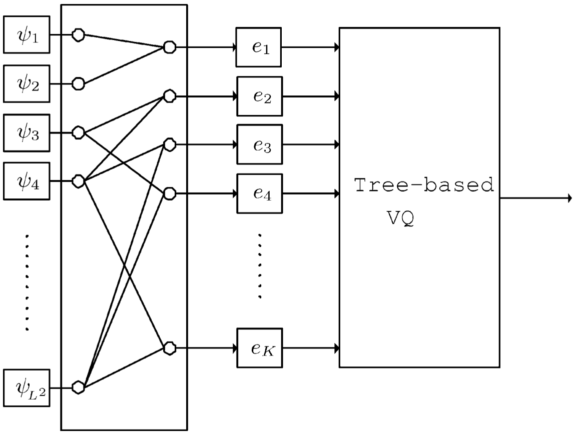


Fig. 2. Proposed two-stage VQ structure.

II. Two-Stage VQ Design

We first describe the MP algorithm, and then present our approach for dictionary approximation and efficient atom extraction.

A. Matching Pursuit (MP) Algorithm

Let $\mathcal{D}$ be a dictionary of over-complete image bases $\{B_j\}$. The MP algorithm decomposes an image into a linear expansion of the bases in the dictionary by a succession of greedy steps. The image, f, is first decomposed into

$$f = \langle f, B_{j_0} \rangle B_{j_0} + Rf$$

where $B_{j_0} = \arg_{B_j \in \mathcal{D}} \max\{|\langle f, B_j \rangle|\}$, and Rf is the residual image after approximating f in the direction of B_{j_0}. The dictionary element, B_{j_0}, combined with the inner product value $\langle f, B_{j_0} \rangle$ is called an atom. The MP algorithm decomposes the residual image Rf by projecting it on to the basis functions of $\mathcal{D}$, as was done for f. After M iterations, an approximation of the image f can be obtained from the M atoms by

$$\tilde{f}_M = \sum_{k=0}^{M-1} \langle R^k f, B_{j_k} \rangle B_{j_k}$$

and $\tilde{f}_M$ converges strongly to f as $M \to \infty$.

B. Dictionary Approximation

In our approach, the MP dictionary is approximated by its eigenfunctions, each of which is then approximated by the DWT with Haar bases. Let the bases in the dictionary $\mathcal{D}$ be $B_1, B_2, \ldots, B_{|\mathcal{D}|}$, which can be non-separable. If the sizes of the bases are different, zeros are added to equalize them. We apply PCA to the bases and select the K eigenfunctions with the largest eigenvalues. Let the K eigenfunctions be denoted as $\mathbf{E}_1, \mathbf{E}_2, \ldots, \mathbf{E}_K$. Then, we have

$$\tilde{B}_b = \sum_{j=1}^{K} \langle B_b, \mathbf{E}_j \rangle \mathbf{E}_j$$

where b is the index of the basis and $b = 1, 2, \ldots, |\mathcal{D}|$.

The eigenfunctions $\{\mathbf{E}_j\}$ are wavelet-transformed using the Haar wavelet, denoted as $\{\psi_{mn}\}$. The Haar wavelet is used because its filtering operations can be implemented efficiently. To further reduce the computational cost, each eigenfunction is approximated by the N largest DWT coefficients, i.e.,

$$\mathbf{E}_i^w = \sum_{(m,n)\in\Psi_i} \beta_{i,mn}\psi_{mn}\ , i = 1, 2, \ldots, K \tag{1}$$

where Ψ_i is the index set of the N DWT coefficients of the ith eigenfunction. Because $\{\mathbf{E}_i^w\}$ is an approximation of $\{\mathbf{E}_i\}$, the orthogonal property of $\{\mathbf{E}_i\}$ does not hold for $\{\mathbf{E}_i^w\}$. Applying the *Gram-schmidt* procedure to $\{\mathbf{E}_i^w\}$, we have

$$[\,\mathbf{E}_1^N\mathbf{E}_2^N\ldots\mathbf{E}_K^N\,] = [\,\mathbf{E}_1^w\mathbf{E}_2^w\ldots\mathbf{E}_K^w\,]\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1K} \\ & a_{22} & a_{23} & \cdots & a_{2K} \\ & & \ddots & & \\ & & & & a_{KK} \end{bmatrix}$$

i.e.,

$$\mathbf{E}_j^N = \sum_{i=1}^{j} a_{i,j}\mathbf{E}_i^w.$$

$\mathbf{E}_j^N, j = 1, 2, \ldots, K$ are orthonormal functions used to approximate the bases $\{B_b\}$. Hence, we can derive the approximation of our dictionary basis as

$$\begin{aligned}\hat{B}_b^N &= \sum_{j=1}^{K}\langle B_b, \mathbf{E}_j^N\rangle\mathbf{E}_j^N \\ &= \sum_{j=1}^{K}\left\langle B_b, \left(\sum_{i=1}^{j} a_{i,j}\mathbf{E}_i^w\right)\right\rangle\mathbf{E}_j^N \\ &= \sum_{j=1}^{K}\left(\sum_{i=1}^{j} a_{i,j}\langle B_b, \mathbf{E}_i^w\rangle\right)\mathbf{E}_j^N \\ &= \sum_{j=1}^{K}\alpha_{b,j}^N\mathbf{E}_j^N \end{aligned}\tag{2}$$

where $\alpha_{b,j}^N$ is the projection of basis B_b onto $\mathbf{E}_j^N$. Because $\{\mathbf{E}_j^N\}$ are orthonormal functions, the norm $\hat{B}_b^N$ is equal to the norm of the vector $[\alpha_{b,1}^N, \alpha_{b,2}^N, \ldots, \alpha_{b,K}^N]$, i.e.,

$$\parallel \hat{B}_b^N \parallel = \left[\sum_{j=1}^{K}(\alpha_{b,j}^N)^2\right]^{1/2}.$$

The inner product between f and the normalized basis $\hat{B}_b^N/\parallel \hat{B}_b^N \parallel$ can be expressed as

$$\begin{aligned}\left\langle f, \frac{\hat{B}_b^N}{\parallel \hat{B}_b^N \parallel}\right\rangle &= \frac{1}{\parallel \hat{B}_b^N \parallel} \times \sum_{j=1}^{K}\langle B_b, \mathbf{E}_j^N\rangle\langle f, \mathbf{E}_j^N\rangle \\ &= \frac{1}{\parallel \hat{B}_b^N \parallel} \times \sum_{j=1}^{K}\alpha_{b,j}^N\left(\sum_{i=1}^{j} a_{i,j}\langle f, \mathbf{E}_i^w\rangle\right) \\ &= \frac{1}{\parallel \hat{B}_b^N \parallel}\sum_{i=1}^{K}\left(\sum_{j=i}^{K}\alpha_{b,j}^N \times a_{i,j}\right)\langle f, \mathbf{E}_i^w\rangle \\ &= \sum_{i=1}^{K}\alpha'_{b,i}\langle f, \mathbf{E}_i^w\rangle \end{aligned}\tag{3}$$

where

$$\alpha'_{b,i} = \frac{\left(\sum_{j=i}^{K}\alpha_{b,j}^N \times a_{i,j}\right)}{\parallel \hat{B}_b^N \parallel}.$$

Hence, the inner product $\langle f, \hat{B}_b^N/\parallel \hat{B}_b^N \parallel\rangle$ can be obtained from the inner product of two K dimensional vectors.

$$\left\langle f, \frac{\hat{B}_b^N}{\parallel \hat{B}_b^N \parallel}\right\rangle = [\,\alpha'_{b,1}\alpha'_{b,2}\ldots\alpha'_{b,K}\,]\begin{bmatrix}\langle f, \mathbf{E}_1^w\rangle \\ \langle f, \mathbf{E}_2^w\rangle \\ \vdots \\ \langle f, \mathbf{E}_K^w\rangle\end{bmatrix} = \bar{\alpha}_b^T\bar{f}. \tag{4}$$

Note that $\bar{\alpha}_b$ can be pre-computed. Moreover, according to (1)

$$\langle f, \mathbf{E}_i^w\rangle = \sum_{(m,n)\in\Psi_i}\beta_{i,mn}\langle f, \psi_{mn}\rangle, i = 1, 2, \ldots, K; \tag{5}$$

$\beta_{i,mn}$ can also be precomputed.

C. Tree-Based VQ

Our atom extraction technique is combined with the calculation of the inner product by a VQ method. Equation (4) shows that the inner product between an MP residual and a basis function can be obtained from the inner product of two vectors, $\bar{f}$ and $\bar{\alpha}_b$. Vector $\bar{\alpha}_b$ only contains the basis information and can be pre-calculated, while vector $\bar{f}$ depends on the MP residual and must be re-computed or updated at each iteration. The vectors $\{\bar{\alpha}_b|\ b = 1, 2, \ldots, |\mathcal{D}|\}$ form $|\mathcal{D}|$ points in the K dimensional vector space. Using the VQ technique, the vector space can be partitioned into $|\mathcal{D}|$ components centered at each $\bar{\alpha}_b$. If $\bar{f}$ lies inside a component whose center corresponds to the basis b^*, then the absolute inner product between $\bar{f}$ and $\bar{\alpha}_{b^*}$ will be the largest among $\bar{f}$ and any vector in $\{\bar{\alpha}_b|\ b = 1, 2, \ldots, |\mathcal{D}|\}$. With VQ, finding the atom from all the inner product values of $\bar{f}$ and the basis functions is the same as locating the nearest $\bar{\alpha}_b$ to $\bar{f}$. The complexity of finding the atom can be further reduced by using tree-based VQ [5]–[7] to find the best codeword. If it cannot do so, it usually finds a codeword whose performance is close to the best codeword.

We use a simple *bottom-up* algorithm to build a binary tree, based on VQ, in which a parent node has two child nodes. Our objective is to organize the codewords, $\bar{\alpha}_b$, in such a way that the binary search algorithm can find the basis whose inner product is close to that obtained by an exhaustive search. Let d be the lowest level of the tree. We use $\bar{\alpha}_b^d (= \bar{\alpha}_b)$ to represent that the codeword $\bar{\alpha}_b$ is at level d. To build the parent level, we find the pair of child nodes that gives the maximum inner product value

$$|\langle\bar{\alpha}_p^d, \bar{\alpha}_q^d\rangle| = \max_{i,j\in\mathcal{D}}|\langle\bar{\alpha}_i^d, \bar{\alpha}_j^d\rangle|.$$

If the inner product $\langle\bar{\alpha}_p^d, \bar{\alpha}_q^d\rangle$ is positive, we use the mean vector of $\bar{\alpha}_p^d$ and $\bar{\alpha}_q^d$ to represent their parent node $\bar{\alpha}_1^{d-1}$; otherwise, the parent node is the mean vector of α_b^d and $-\alpha_q^d$. By the same procedure, we select another pair of child nodes from the remaining vectors in $\{\bar{\alpha}_b^d|\ b = 1, 2, \ldots, |\mathcal{D}|\} - \{\bar{\alpha}_p^d, \bar{\alpha}_q^d\}$ and construct their parent node $\bar{\alpha}_2^{d-1}$. We continue the procedure until all the vectors in $\{\bar{\alpha}_b^d|\ b = 1, 2, \ldots, |\mathcal{D}|\}$ have been selected and the

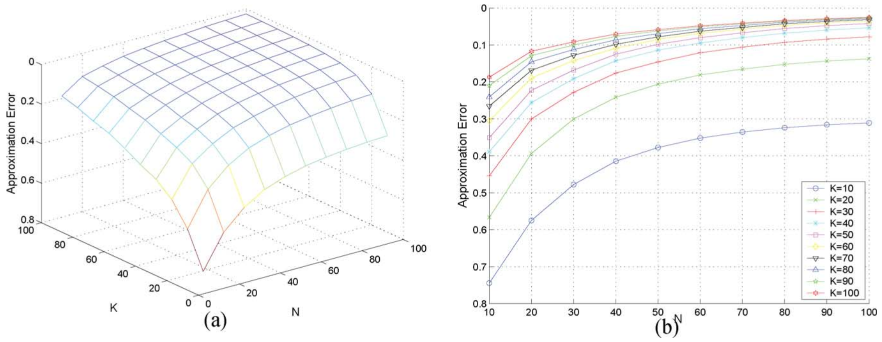


Fig. 3. (a) Approximation error as a function of N and K. (b) Approximation error versus N curves for different K. Larger N and K yield a better approximation of the Gabor dictionary. The error rate increases slowly as N and K decrease. When both N and K are less than 25, MSE increases significantly.

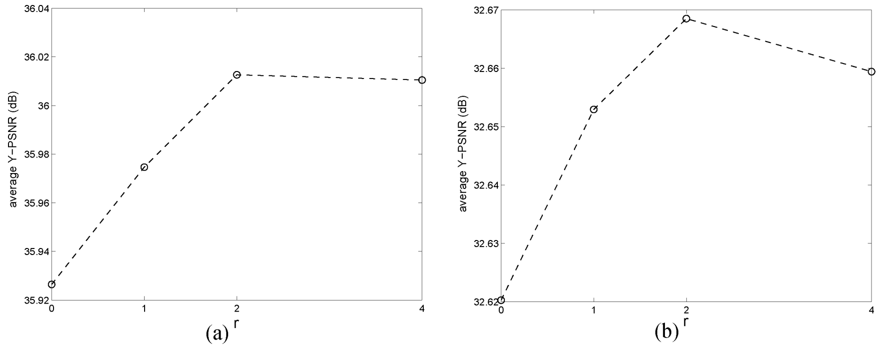


Fig. 4. Average Y-PSNRs of various N and K, whose ranges are between 10 and 100, with different merging thresholds, $\{\sqrt{\mathrm{MSE}/r} | r = 0, 1, 2, 4\}$. (a) Akiyo sequence. (b) Mother and Daughter sequence encoded at 30 kbps, 10 fps.

upper level $d-1$ has been constructed. By repeating the above procedure, we can build the $d-k$ level from the nodes in the $d-k+1$ level until the root node is reached. The tree will then be balanced with a depth of $d\ (= \log_2 |\mathcal{D}|)$.

To query a codeword in the tree, we use a *top-down* approach. If the current internal node is $\bar{\alpha}_j^k$, and its left and right children are respectively $\bar{\alpha}_p^{k+1}$ and $\bar{\alpha}_q^{k+1}$, then node $\bar{\alpha}_p^{k+1}$ will be selected if

$$|\langle \bar{f}, \bar{\alpha}_p^{k+1} \rangle| \; > \; |\langle \bar{f}, \bar{\alpha}_q^{k+1} \rangle|;$$

otherwise, node $\bar{\alpha}_q^{k+1}$ will be selected. This procedure is repeated at each encountered internal node until a leaf of the tree is reached. Our binary tree-based VQ requires $\mathcal{O}(K \times \log_2 |\mathcal{D}|)$ to search the basis vector for $\bar{f}$. Although the search procedure does not always find the basis that gives the maximum absolute inner product, our experiments show that the probability of finding a basis with an absolute inner product close to that value is high. The effectiveness of our binary tree-based VQ will be discussed in Section IV.

D. Bases Elimination

The MP algorithm is designed to greedily reduce the distortion at each iteration; however, the approach may not be the best for compression purposes, because the coding objective is to select the atom that gives the largest rate-distortion reduction at each iteration [20]. Reducing the number of bases in a dictionary increases the distortion, but it reduces the number of bits required to encode the bases' indices. Therefore, the coding performance may be improved. The number of bases in the proposed method depends on two parameters: the number of eigenfunctions K, and the number of dyadic wavelet coefficients N. The mean square error (MSE) of our dictionary approximation is measured by

$$\mathrm{MSE}(N,K) = \frac{1}{|\mathcal{D}|} \sum_{b=1}^{|\mathcal{D}|} \left\| \frac{\hat{B}_b^N}{\| \hat{B}_b^N \|} - B_b \right\|^2.$$

Although we can approximate any MP dictionary, we use the Gabor dictionary in [14] as our target dictionary. Fig. 3 shows the MSE of Gabor dictionary approximation as a function of N and K. When both N and K are small, some similarly shaped bases can be clustered so that only a representative basis is kept to represent the others, which reduces the number of bases. We cluster bases according to the following rule: if the Euclidean distance between two bases is less than a given threshold, we only keep one of them. Reducing the size of a dictionary will reduce the complexity of computing the inner products as well as the entropy of encoding the indexes of the bases. Fig. 4 shows the average Y-PSNRs of various N and K with different merging thresholds, $\{\sqrt{\mathrm{MSE}/r} |\ r = 0, 1, 2, 4\}$; $r = 0$ means that dictionary elimination processing is not applied. As shown in the figure, by clustering the bases, the Y-PSNR is slightly better than that without dictionary elimination. Based on the experiment results, we set $r = 2$ in the following experiments, which achieves the best PSNR gain. The increase in the Y-PSNR after basis elimination is mainly because fewer bits are used to encode a basis's index of each atom. Fig. 5(a) shows the number of bases in the reduced Gabor dictionary as a function of N and K when the threshold is $\sqrt{\mathrm{MSE}/2}$, while Fig. 5(b) shows that the average number of bits used to encode a basis decreases as the approximation error increases.

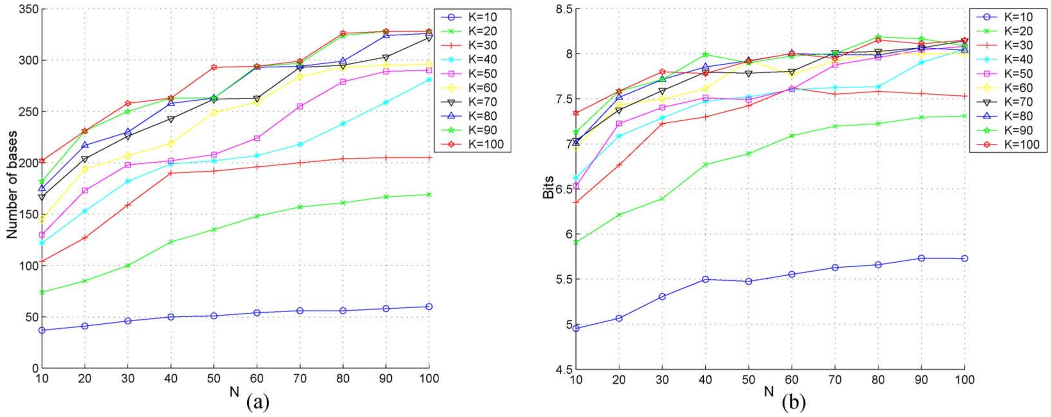


Fig. 5. (a) Number of bases in the reduced dictionary as a function of N and K. (b) Average number of bits used to encode the index of a basis of various N and K for the Foreman sequence at 30 kbps.

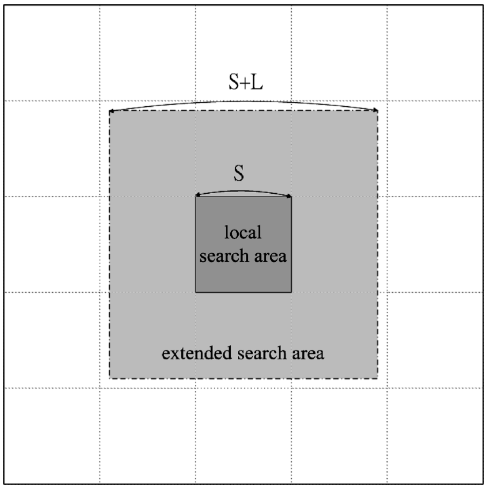


Fig. 6. The size of the local search area (middle box) is S by S, and that of the extended search area (outer box) is $S+L$ by $S+L$.

III. Complexity Analysis

We now analyze the complexity of our approach and compare it with that of other approaches. There are two different implementations of the MP algorithm. One recalculates all the inner products at each iteration, while the other calculates all the inner products once at the beginning, and modifies them according to pre-calculated update information. The latter approach, called the MP update algorithm, is faster at the expense of needing more storage space for the update information. Because finding an atom from within an entire image is very time-consuming, we use the popular suboptimal algorithm [14], [1], which divides an MP residual into disjoint blocks of size S by S and finds an atom within the block that has the highest (weighted) energy.

In Section III-A, we analyze the complexity of the implementation that recalculates all the inner products at each iteration. Then, in Section III-B, we analyze the complexity of using the MP update algorithm.

A. Without Using MP Update

First, we address the complexity of the proposed two-stage VQ approach without using the MP update algorithm. We measure the complexity of calculating (5) for a residual block and the complexity of our tree search algorithm. We also compare the complexity of finding an atom with that of the algorithm proposed in [16] and [18].

1) Inner Product of an MP Residual and Eigenfunctions: To compute the inner products between a residual block and the eigenfunctions, we first compute the DWT of each sub-block of size L by L centered at a pixel in the S by S region, as shown in Fig. 6. We apply the undecimated 2-D wavelet transform (*à trous* algorithm) [17] to obtain all the DWT coefficients. A level of the 2-D undecimated Haar filter bank takes $6(S+L)^2$ additions to decompose a size $S+L$ by $S+L$ image block. There are $\log_2 L$ levels of decomposition for an L by L block. Thus, the total number of additions is

$$\Gamma_W = 6 \times \log_2 L \times (S+L)^2. \tag{6}$$

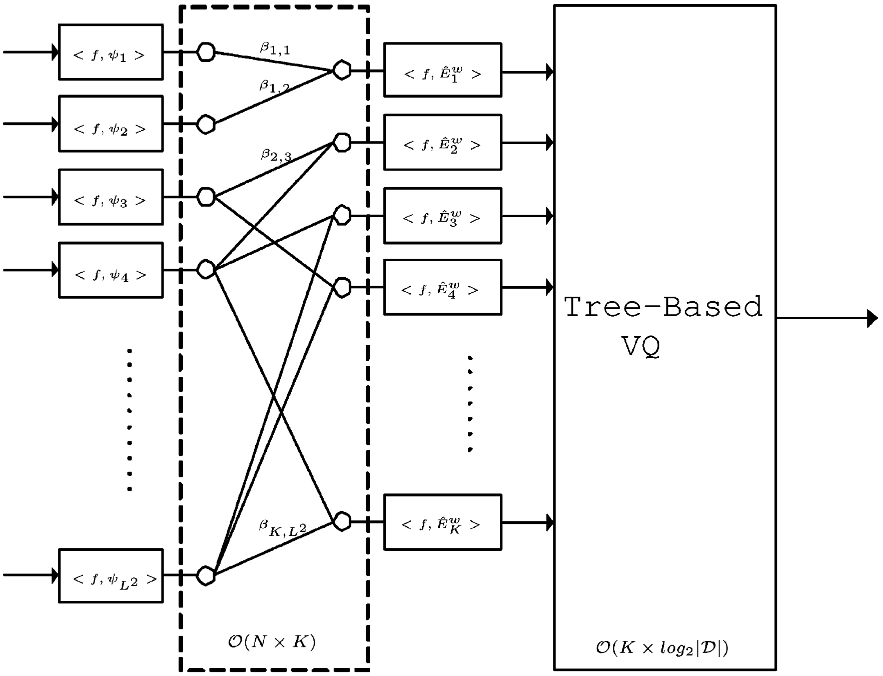
Fig. 7 illustrates the block diagram of the implementation of our approach. In the figure, the weights in the box were pre-computed according to (1). Because the number of dyadic wavelet coefficients required to approximate each eigenfunction is N, the complexity of calculating the inner products of K eigenfunctions in the local search area S^2 is $\Gamma_E = K \times N \times S^2 \ (\text{adds} + \text{mults})$. If we use *op* to denote an addition or a multiplication, we then have

$$\Gamma_E = 2 \times K \times N \times S^2 \ (\text{ops}). \tag{7}$$

2) Tree Search and Comparison: The last step in Fig. 7 applies a binary tree search to find a basis. The complexity of finding the basis in the search block is

$$\begin{aligned} \Gamma_S &= 2 \times \log_2 |\mathcal{D}| \times K \times S^2 \ (\text{adds} + \text{mults}) \\ &= 4 \times \log_2 |\mathcal{D}| \times K \times S^2 \ (\text{ops}). \end{aligned} \tag{8}$$

If the implementation is used without the proposed tree-based VQ, it is necessary to form all the inner products between the residual image and the basis from the combination of the inner products between the residual image and the eigenfunctions, and then find the maximum atom by exhaustive search. This would require $|\mathcal{D}| \times K \times S^2 \ (\text{adds}+\text{mults}) = 2 \times |\mathcal{D}| \times K \times S^2 \ (\text{ops})$ to obtain all the inner products and $|\mathcal{D}| \times S^2$ comparisons to find the atom by exhaustive search. The advantage of using tree-



(b)

Fig. 7. Block diagram of the implementation of our proposed structure. The weights in the large left-hand box are pre-calculated and the number of dyadic wavelet coefficients used to approximate each eigenfunction $N = 2$. The input data is f, which is first applied to the DWT with Haar bases to obtain the wavelet coefficients of the data $\{\langle f, i\rangle\}$. The complexity of computing all wavelet coefficients is $\mathcal{O}(6 \times \log_2 L)$.

TABLE I
COMPLEXITY OF THE TRADITIONAL TWO-STAGE APPROACH AND OUR TWO-STAGE VQ APPROACH MEASURED PER ITERATION

Traditional two-stage approach	Proposed two-stage approach
$M \times A \times (S+L)^2$(elementary)	$6 \times \log_2 L \times (S+L)^2$(DWT with input)
$G \times \lvert\mathcal{D}\rvert \times S^2$ (approximated bases)	$2 \times K \times N \times S^2$ (eigenfunctions)
$\lvert\mathcal{D}\rvert + \lvert\mathcal{D}\rvert S^2$(maximum atom)	$4 \times K \times log_2\lvert\mathcal{D}\rvert \times S^2$ (maximum atom)

based VQ is obvious and becomes more significant as the size of a dictionary increases.

3) Comparison Based on One Iteration: Fig. 1(a) shows the structure of a conventional two-stage approach, in which a basis is approximated by an elementary dictionary and its previous basis. If the dictionary consists of A elements and the average number of operations needed to compute the elementary inner product is M, then the complexity of computing the inner products of the elementary dictionary over the extended search region will be $M \times A \times (S+L)^2$ (ops). To approximate the bases, each basis is assumed to be approximated by a linear combination of the average of G different elements and other bases. It takes $G \times |\mathcal{D}| \times S^2$ (adds) to compute the inner products of the bases. Note that, because the bases are approximated, their norms are not equal to one. It requires $|\mathcal{D}| \times S^2$ comparisons and extra $|\mathcal{D}|$ (mults) to normalize them in order to extract the maximum atom. We summarize the complexity of the two approaches, measured per iteration, in Table I.

As shown in the table, each approach has different parameters; therefore, it is difficult to compare their complexity precisely. However, our approach is systematic, whereas the traditional approach uses ad hoc methods to find elementary functions and the computational order. Thus, designing a fast MP encoder using our approach is simpler than using a traditional approach when a dictionary is large. Also, the complexity of the traditional approach, $\mathcal{O}(M \times A + (G+1) \times |\mathcal{D}|)$, is a linear function of the dictionary size; while that of our approach, $\mathcal{O}(K \times N + K \times \log_2 |\mathcal{D}|)$, is a logarithmic function of the dictionary size. This also gives our approach an advantage if the goal is to approximate a large dictionary.

B. Using MP Update

The complexity analyzed so far is based on recalculating the inner products at each iteration. Since the MP update algorithm reduces the computational complexity at the cost of using extra memory to store the update information, in the following, we use the algorithm to further reduce the computational complexity of our approach. At each iteration, we obtain the vector $[\langle R^k f, \mathbf{E}_i^w\rangle]$, $i = 1, \ldots, K$ and use it to find an atom in tree-based VQ. We then update the vector to $[\langle R^{k+1} f, \mathbf{E}_i^w\rangle]$ at each iteration according to

$$\langle R^{k+1} f, \mathbf{E}_i^w\rangle = \langle R^k f, \mathbf{E}_i^w\rangle - \langle R^k f, \hat{B}'_{j_k}\rangle\langle \hat{B}'_{j_k}, \mathbf{E}_i^w\rangle. \quad (9)$$

In the above equation, $\langle \hat{B}'_{j_k}, \mathbf{E}_i^w\rangle$, which is independent of each iteration, is pre-calculated and stored; and $\langle R^k f, \hat{B}'_{j_k}\rangle$ is the inner product of the atom of the previous iteration. We store all the inner products of $\{\langle R^k f, \mathbf{E}_i^w(x,y)\rangle\}$ (of size $K \times S^2$ real values) so that the update step can be implemented efficiently

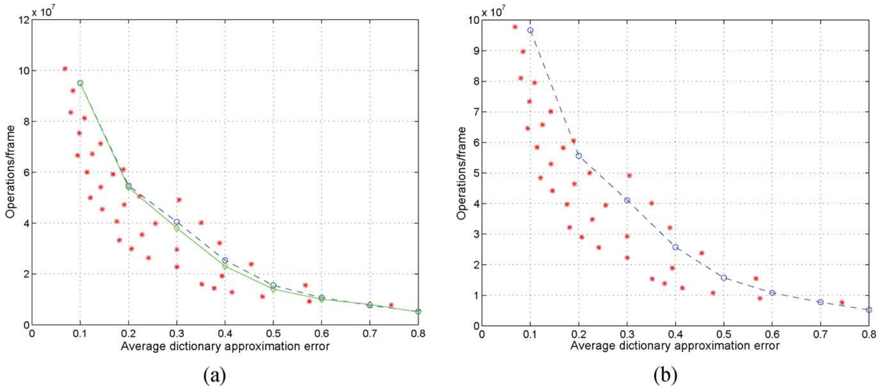


Fig. 8. Comparison of the complexity of our approximating dictionary and that in [16] measured as average operations per frame for 10 S. (a) QCIF Coast guard sequence encoded at 48 kbps, $N_{\text{atom}} = 170$ and $M_{\text{blk}} = 30$. (b) QCIF Foreman sequence, encoded at 48 kbps, $N\text{atom} = 173$ and $\text{Mblk} = 28$. The solid curve of Neff *et al.* is taken from [16] and the dashed curves are estimated by (12). The stars indicate the results of our method for different N and K.

using one addition and one multiplication at each iteration with complexity:

$$\Gamma_{UE} = K \times S^2 \ \ (\text{adds} + \text{mults}) = 2KS^2 \ (\text{ops}). \qquad (10)$$

Let us assume that, on average, M_{blk} blocks are encoded in one frame. To find the first atom of each block, we need to calculate all the inner products within that block. As discussed in Sections III-A.I and III-A.II, it takes $\Gamma_W + \Gamma_E + \Gamma_S$ ops to find the first atom of a block, where Γ_W, Γ_E, and Γ_S are defined in (6)–(8) respectively. The update algorithm can then be applied to the other atoms. Therefore, after N_{atom} iterations, the complexity is

$$M_{\text{blk}} \times (\Gamma_W + \Gamma_E) + (N_{\text{atom}} - M_{\text{blk}}) \times \Gamma_{UE} + N_{\text{atom}} \times \Gamma_S. \quad (11)$$

Compared to the two-stage approach with the update algorithm in [16], the complexity of N_{atom} is

$$\Gamma(IF) + N_{\text{atom}} \times (\Gamma(FA) + \Gamma(EU)) \qquad (12)$$

where $\Gamma(IF)$ is the initial step of each frame, $\Gamma(FA)$ is the complexity of finding the atom in each iteration, and $\Gamma(EU)$ is the complexity of applying the update. For detailed derivations of these items, readers should refer to [16].

We compare the complexity (ops/frame) of our approach with that given in [16]. Fig. 8 compares the operations per frame of different methods. The curves of Neff *et al.* are either taken from their paper or estimated according to (12). We performed the estimation by first obtaining $\Gamma(FA)$, $\Gamma(EU)$, and $\Gamma(IF)$ from the corresponding figures and the table in [16]. Then, using these values and N_{atom}, obtained by our simulation, we calculated the complexity of the approach in [16] from (12). Note that our estimated complexity is very close to that given in [16]. As can be observed from the figure, for most N and K, the complexity of our algorithm is lower when both methods have the same dictionary approximation error.

IV. Performance Evaluation

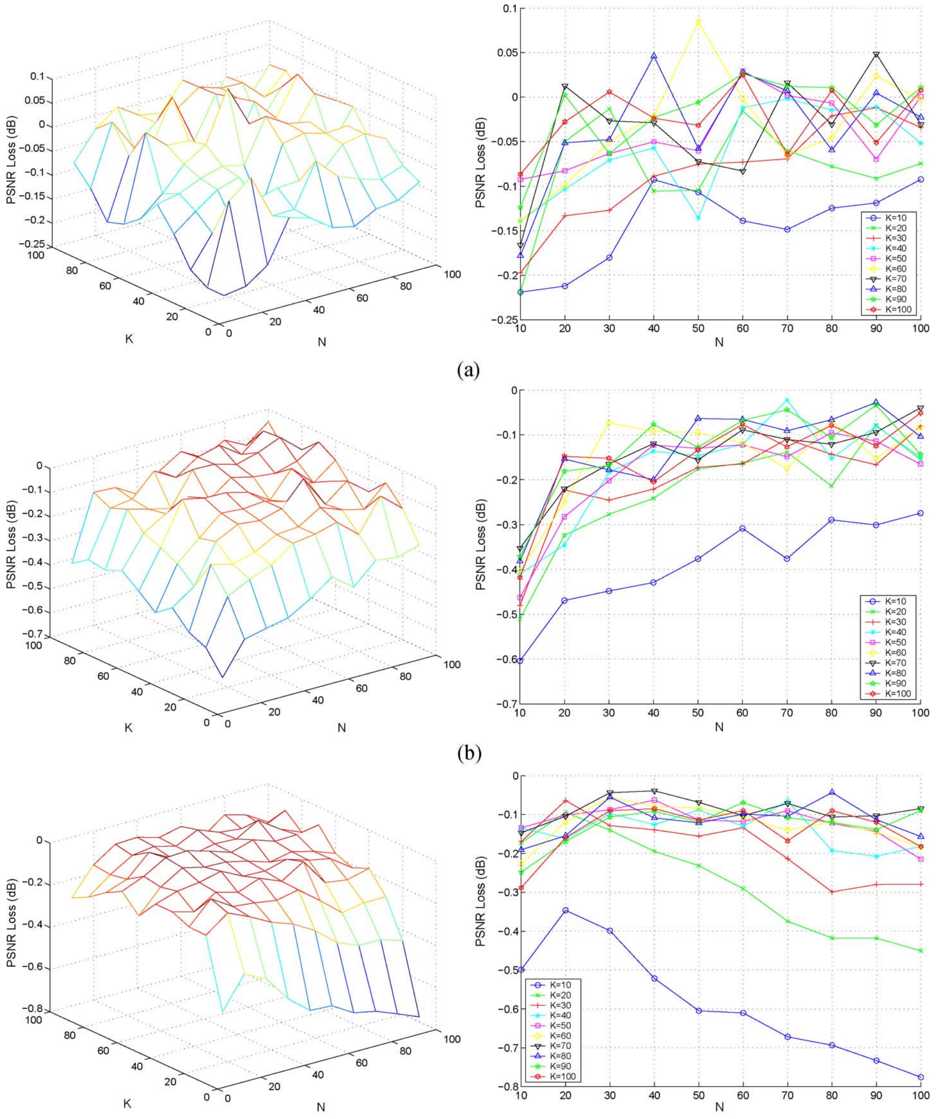
Here, we evaluate the coding performance of our two-stage VQ algorithm. The parameters of an atom are the index, the position of the basis, and the inner product value. We use adaptive arithmetic coding to encode the indexes of the bases. The inner product values of the bases are encoded by a bit plane based approach, whereby crucial atom positions are encoded based on a quadtree representation of a bit plane [10]. In the following experiments, we use the most popular separable Gabor dictionary with 400 bases [14] as our target dictionary. The unrestricted motion vector mode and advanced prediction mode in the H.263 standard [9] are used to obtain the motion vectors. The first frame of a video sequence is an intra-frame (I-frame) encoded by the DCT, while all other frames are inter-frames (P-frames). The sizes of our test sequences are in QCIF format and the testing frame rate is 10 fps.

We measure the efficiency of our approach by the Y-PSNR loss and speed-up factors. The former is defined as the Y-PSNR of the target dictionary minus that of the approximating dictionary. Fig. 9 shows the Y-PSNR loss as a function of N and K at different bit rates for various sequences. As shown in the figure, the Y-PSNR loss increases as N and K decrease. For some N and K, the Y-PSNR of our approximating dictionary at low bit rates of the News sequence is better than that of the original dictionary. This is because the elimination of bases reduces the bits required to encode the indexes of bases. Therefore, the approximating dictionary can encode more atoms than the original dictionary at the same bit rate, which further reduces the distortion.

The reduction in computing time is measured by the speed-up factor, which is obtained by dividing the complexity of the algorithm in [14] by that of our algorithm. Fig. 10 shows the performance versus the speed-up factor. Each point corresponds to a pair of N and K. The envelope curves indicate the best performance that our algorithm can achieve as a function of the speed-up factor. The general trend of the curves indicates that increasing the speed-up factor degrades the performance. Note that, for all the sequences, there are some (N, K) that speed up the computation more than 50 times with a Y-PSNR loss of approximately 0.1 dB.

So far, we have demonstrated the case of finding the optimal pair of N and K that yields a given performance loss without resource constraints. For resource-constrained environments, we are interested in determining the optimal pair for a given constraint. Our method is described in Appendix 1.

To compare the subjective quality of our approach with that of the 2-D Gabor dictionary, we used the methodology of



(a)

(b)

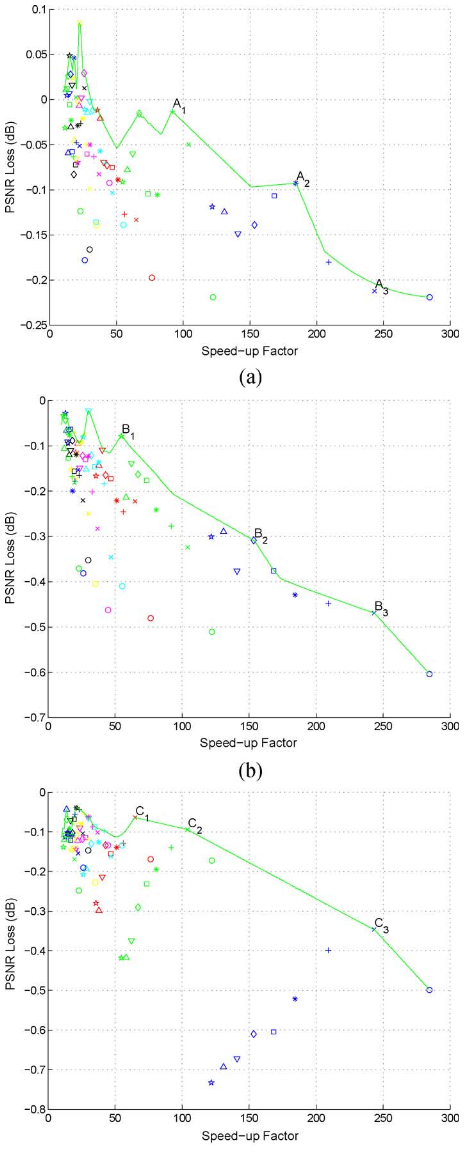
(c)

Fig. 9. Average Y-PSNR loss of our approach. The left subfigures are functions of N and K, which are both between 10 and 100, and the right subfigures are functions of N for different K. The average Y-PSNRs of the Gabor dictionary for the News, Coast guard, and Stefan sequences are 30.68, 27.25, and 28.97 dBs, respectively. (a) News sequence at 20 kbps. (b) Coast guard sequence at 31 kbps. (c) Stefan sequence at 128 kbps.

subjective assessment in [8]. The double-stimulus impairment scale (DSIS) was used to evaluate the subjective quality. In this test procedure, participants were shown multiple pairs of sequences. Each pair consisted of an original sequence and a compressed sequence, both of which were rather short. The original sequence was presented first, followed by a gray period, then the compressed sequence was presented. Both sequences were presented twice. The participants were required to score the sequences using a five-grade impairment scale: imperceptible (5), perceptible, but not annoying (4), slightly annoying (3), annoying (2), and very annoying (1). The subjective evaluation results are shown in Fig. 11. For each point, the Y axis represents the average DSIS score given by 40 participants for the sequences encoded using the Gabor dictionary, and the X axis represents the average score for the sequences encoded using our two-stage VQ dictionary with different speed-up factors. As shown in the figure, the points are scattered around the diagonal line, indicating that the two approaches achieve the same score. From the test, we conclude that our approach does not degrade the perceptual quality of an image, even if the Gabor dictionary is approximated with speed-up factors of up to 250. Snapshots of a video sequence encoded using the Gabor dictionary and our two-stage VQ dictionary with different speed-up factors are shown in Fig. 12. The overall perceptual quality of the pictures is comparable. Specifically, the artifacts induced by reducing the complexity are noisy spots on the faces. This is due to the irregularity of the approximating bases when the approximation error is large, which corresponds to a large speed-up factor.

V. Conclusion

We have proposed a two-stage VQ approach that combines dictionary approximation and atom extraction in a new framework. Compared to previous two-stage dictionary approximations, our approach is systematic and has a lower computational complexity when the size of the target dictionary



(a)

(b)

(c)

Fig. 10. Average Y-PSNR loss and corresponding speed-up factor for various test sequences of different N and K; each point corresponds to a pair of N and K, whose ranges are between 10 and 100. Note that the speed-up of News, Coast guard, and Stefan achieved by our algorithms is respectively 190, 60, and 110 with negligible 0.1 PSNR loss compared to that in [14]. The curves are the upper envelopes of the points. (a) News sequence at 20 kbps. (b) Coast guard sequence at 31 kbps. (c) Stefan sequence at 113 kbps.

is large. We applied our method to approximate the Gabor dictionary, and showed the trade-off between performance loss and the speed-up in computational time. Through examples, we demonstrated that our method can achieve a complexity reduction factor of up to 100 in exchange for approximately 0.1-dB performance loss. Subjective evaluations indicate that our approach retains the perceptual quality of a sequence, even when the speed-up factor is as high as 250. We have also proposed an approach for obtaining the best coding performance within the constraint of a given complexity. We used MSE as the criterion for bases elimination. However, using a perceptual measurement to merge bases would be an interesting research topic worthy of further study.

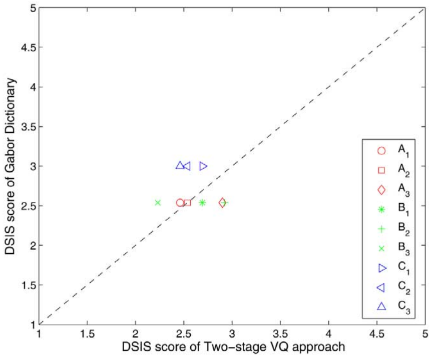


Fig. 11. Average DSIS scores of the Gabor dictionary and our two-stage VQ approach for sequences with various speed-up factors. The scores are the averages of 40 participants. Points A_1, A_2, A_3 are from News; B_1, B_2, B_3 are from Coast guard; and C_1, C_2, C_3 are from Stefan. The parameters of the points are indicated in Fig. 10.

APPENDIX
RESOURCE-CONSTRAINED PARAMETER SELECTION

The computational efficiency of the proposed method depends on the number of eigenfunctions, K, and the number of dyadic wavelet coefficients, N, used to approximate an eigenfunction. Determining N and K values for the best possible coding performance with constrained encoder resources is a crucial step in adapting the proposed algorithm to a heterogeneous environment in which encoder resources vary. We thus propose an approach for optimizing parameters that generates the least Y-PSNR loss according to the constraints of an encoder's resources. Let $dP(N,K)$ be the Y-PSNR loss of our algorithm if N and K are chosen as the algorithmic parameters; and let $\mathbb{C}$ be the affordable computational cost for a system to execute the MP algorithm. $\mathbb{C}$ depends on the system's resources, such as the CPU and memory constraints at the encoder side. We formulate the following constrained optimization problem

$$\begin{cases}\min & dP(N,K)\\ & C(N,K)\le \mathbb{C}\end{cases} \tag{13}$$

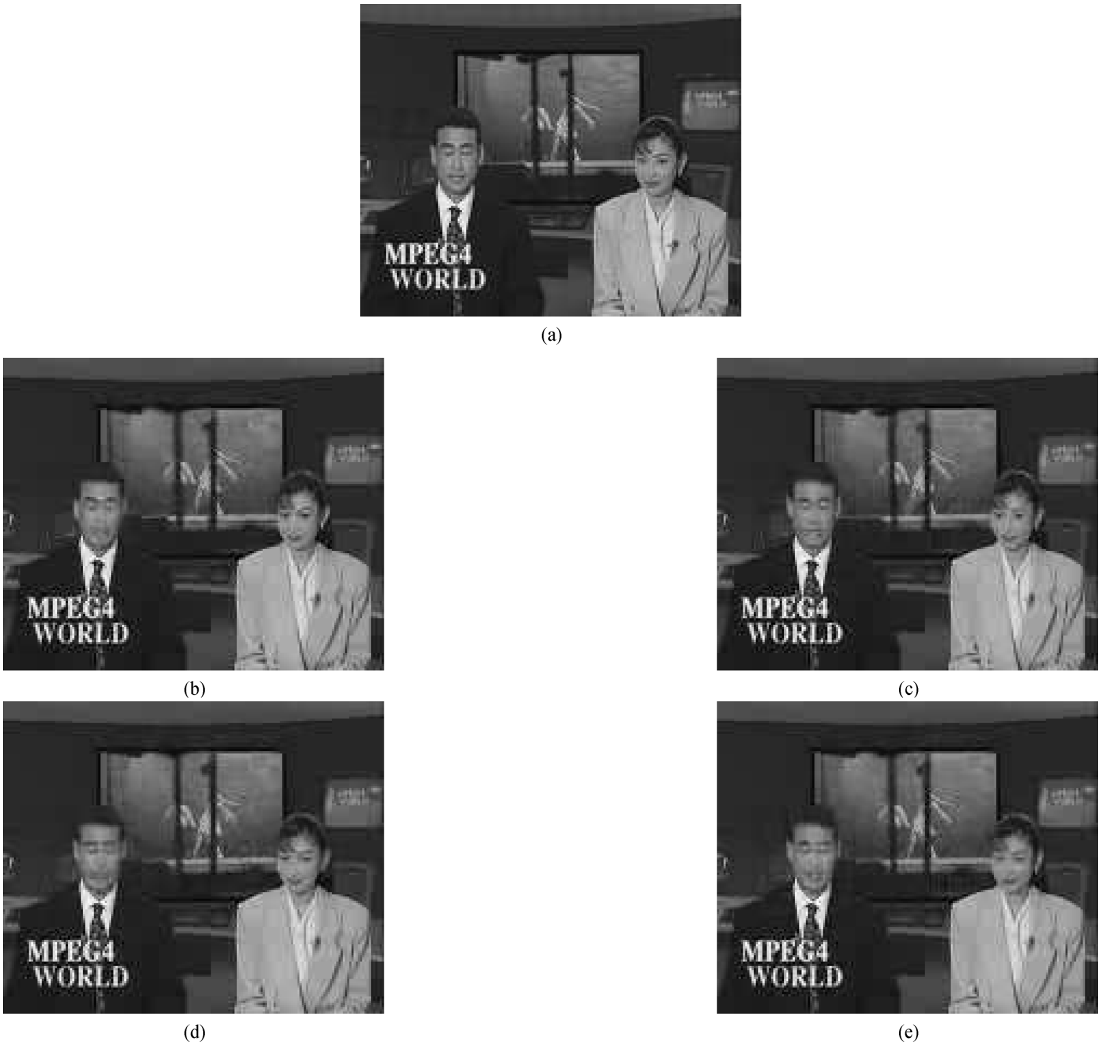
where $C(N,K)$ is the computational cost of executing the MP algorithm using our proposed dictionary with a given N and K. We use the following example to demonstrate the concept of our approach. For simplicity, we approximate the average Y-PSNR loss of executing eleven MPEG-4 QCIF sequences at 30 kbps as the quadratic equation

$$-dP(N,K)=a_1K^2+a_2K\ N+a_3N^2+a_4K+a_5N+a_6. \tag{14}$$

The manifold shown in Fig. 13 is generated by fitting the above equation. For a case where the MP update algorithm is not applied, the computational cost per atom is

$$C(N,K)=c_1K\times N+c_2K\times\log_2|\mathcal{D}|+6(L+S)^2\log_2 L+Y \tag{15}$$

where Y is a system constant; and $6(L+S)^2\log_2 L$, $c_1K\times N$, and $c_2K\times\log_2|\mathcal{D}|$ are respectively the cost per atom of: 1) the DWT; 2) obtaining the inner products between the residual image and the eigenfunctions; and 3) tree-based VQ. Because $dP(N,K)$ is a monotonically decreasing function of N and K, and $C(N,K)$ is a monotonically increasing function of the



(a)

(b)

(c)

(d)

(e)

Fig. 12. Perceptual quality of Frame 66 of the QCIF News sequence encoded at 20 kb, 10 fps. (a) Original frame. (b) Gabor dictionary. (c) Our approach with a speed-up factor of 92. (d) Our approach with a speed-up factor of 184;. (e) Our approach with a speed-up factor of 243.

parameters, for a given $\mathbb{C}$, the solution that minimizes dP is derived when $C(N,K)=\mathbb{C}$. The above constrained problem can be solved by the Lagrange multiplier approach; that is, by finding the N^* and K^* that minimize the following equation:

$$\min_{N,K} J(N,K) = dP(N,K) + \lambda(C(N,K) - \mathbb{C}) \qquad (16)$$

where λ is the Lagrange multiplier. Note that $C(N,K)$ is the measured cost, while $dP(N,K)$ is the estimated Y-PSNR loss. By partially differentiating $J(N,K)$ with respect to N, K, and λ, we obtain

$$\begin{cases} \frac{\partial J}{\partial K} &= 2a_1K + a_2N + a_4 = 0 \\ \frac{\partial J}{\partial N} &= a_2K + 2a_3N + a_5 = 0 \\ \frac{\partial J}{\partial \lambda} &= c_1K \times N + c_2K \times \log_2 |\mathcal{D}| + 6(L+S)^2 \log_2 L + Y \\ &= \mathbb{C}. \end{cases} \qquad (17)$$

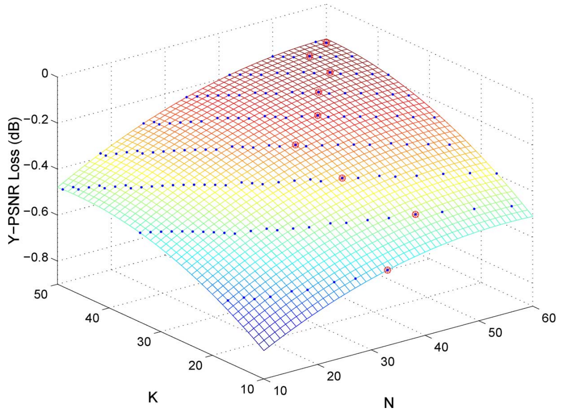


Fig. 13. Numerical optimal solutions for different $\mathbb{C}$ at 30 kbps. All the points on a curve have equal $\mathbb{C}$, but the highlighted circled points have the best Y-PSNR performance. The manifold was generated by averaging eleven MPEG-4 sequences in QCIF format.

The highlighted circled points marked by a star in Fig. 13 are the numerical solutions of N^*, K^*, and λ^* for different $\mathbb{C}$ values.

Acknowledgment

The authors would like to thank the reviewers for their valuable comments which helped us improve the presentation of the manuscript.

References

[1] O. Al-Shaykh, E. Miloslavsky, T. Nomura, R. Neff, and A. Zakhor, "Video compression using matching pursuits," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 1, pp. 123–143, Feb. 1999.

[2] K.-P. Cheung and Y.-H. Chan, "A fast two-stage algorithm for realizing matching pursuit," in *Proc. IEEE ICIP*, 2001, pp. 431–434.

[3] P. Czerepiński, C. Davies, N. Canagarajah, and D. Bull, "Matching pursuits video coding: Dictionaries and fast implentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 7, pp. 1103–1115, Oct. 2000.

[4] K. Engan, S. O. Aase, and J. H. Hus, "Multi-frame compression: Theory and design," *Signal Process.*, vol. 80, no. 10, pp. 2121–2140, Oct. 2000.

[5] W. C. Fang, C. Y. Chang, and B. J. Sheu, "VLSI systolic binary tree-searched vector quantizer for image compression," *IEEE Trans. Very Large Scale Integr. (VLSI) Systems*, vol. 2, no. 1, pp. 33–44, Mar. 1994.

[6] A. Gersho, "On the structure of vector quantizers," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 157–166, Mar. 1982.

[7] R. M. Gray and H. Abut, "Full search and tree searched vector quantization of speech waveforms," in *Proc. IEEE ICASSP*, Paris, France, May 1982, pp. 593–596.

[8] *Methodology for the Subjective Assessment of the Quality of Television Pictures*, ITU-R Recommendation BT.500-11, 2002.

[9] *Video Coding for Low Bit Rate Communication*, ITU-T Recommendation H.263, 1997.

[10] J. L. Lin, W. L. Hwang, and S. C. Pei, "SNR scalability based on bit-plane coding of matching pursuit atoms at low bit rates: Fine-grained and two-layer," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 1, pp. 3–14, Jan. 2005.

[11] J. L. Lin, W. L. Hwang, and S. C. Pei, "Multiple blocks matching pursuit update algorithm for low bit rate video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 3, pp. 331–337, Mar. 2006.

[12] G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[13] R. Neff, T. Nomura, and A. Zakhor, "Decoder complexity and performance comparison of matching pursuit and DCT-based MEPG-4 video codecs," in *Proc. IEEE ICIP*, 1998, pp. 783–787.

[14] R. Neff and A. Zakhor, "Very low bit rate video coding based on matching pursuits," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 158–171, Feb. 1997.

[15] R. Neff and A. Zakhor, "Modulus quantization for matching pursuit video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 6, pp. 895–912, Sep. 2000.

[16] R. Neff and A. Zakhor, "Matching pursuit video coding-part I: Dictioanry approximation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 1, pp. 13–26, Jan. 2002.

[17] H. W. Park and H. S. Kim, "Motion estimation using low-band-shift method for wavelet-based moving-picture coding," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 577–587, Apr. 2000.

[18] D. W. Redmill, D. R. Bull, and P. Czerepiński, "Video coding using a fast non-separable matching pursuits algorithm," in *Proc. IEEE ICIP*, 1998, pp. 769–773.

[19] K. Rose and S. L. Regunathan, "Toward optimality in scalable predictive coding," *IEEE Trans. Image Process.*, vol. 10, no. 7, pp. 965–976, Jul. 2001.

[20] T. Ryen, G. M. Schuster, and A. K. Katsaggelos, "A rate-distortion optimal coding alternative to matching pursuit," in *Proc. IEEE ICASSP*, 2002, pp. 2177–2180.

[21] X. Tang and A. Zakhor, "Matching pursuits multiple description coding for wireless video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 566–575, Jun. 2002.

[22] C. De Vleeschouwer and B. Macq, "Subband dictionaries for low-cost matching pursuits of video residues," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 7, pp. 984–993, Oct. 1999.

[23] C. De Vleeschouwer and B. Macq, "SNR scalability based on matching pursuits," *IEEE Trans. Multimedia*, vol. 2, no. 4, pp. 198–208, Dec. 2000.

[24] C. De Vleeschouwer and A. Zakhor, "Atom modulus quantization for matching pursuit video coding," in *Proc. IEEE ICIP*, Jun. 2002, vol. 3, pp. 681–684.

Jian-Liang Lin was born in I-Lan, Taiwan, R.O.C., in 1975. He received the B.S. degree in electronic engineering from Fu Jen Catholic University, Taipei, Taiwan, R.O.C., in 1997, the M.S. degree in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan, R.O.C., in 1999, and the Ph.D. degree in communication engineering from National Taiwan University, Taipei, Taiwan, R.O.C., in January 2006. He was also a visiting graduate student at the University of Washington, Seattle, from March 2004 to March 2005.

From October 1999 to December 2005, he worked as a Research Assistant at the Institute of Information Science, Academia Sinica, Taipei, Taiwan, R.O.C. Four of those years were credited as military service, which is mandatory in Taiwan. In 2006, he was appointed as a Distinguished Postdoctoral Scholar by the Institute. At the end of 2006, he joined the High Tech Computer (HTC) Corporation, Taoyuan, Taiwan, R.O.C., as a Leader Engineer. Since August 2007, he has been an Assistant Professor in the Institute of Communications Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan, R.O.C. His research interests include matching pursuits, image and video compression, as well as multimedia compression and transmission.

Wen-Liang Hwang (SM'03) received the B.S. degree in nuclear engineering from the National Tsing-Hua University, Hsinchu, Taiwan, R.O.C., the M.S. degree in electrical engineering from the Polytechnic Institute of New York, and, in 1993, the Ph.D. degree in computer science from New York University.

He was a Postdoctoral Researcher with the Department of Mathematics, University of California, Irvine, in 1994. In January 1995, he became a member of the Institute of Information Science, Academia Sinica, Taipei, Taiwan, where he is currently a Research Fellow. He is coauthor of the book *Practical Time-Frequency Analysis* (Academic, 1998). His research interests include wavelet analysis, signal and image processing, and multimedia compression and transmission.

Dr. Hwang was awarded the Academia Sinica Research Award for Junior Research in 2001.

Soo-Chang Pei (SM'89–F'00) was born in Soo-Auo, Taiwan, R.O.C., in 1949. He received the B.S.E.E. degree from the National Taiwan University (NTU), Taipei, Taiwan, R.O.C., in 1970, and the M.S.E.E. and Ph.D. degrees from the University of California, Santa Barbara (UCSB), in 1972 and 1975, respectively.

He was an Engineering Officer in the Chinese Navy Shipyard from 1970 to 1971. From 1971 to 1975, he was a Research Assistant with UCSB. He was the Professor and Chairman in the Electrical Engineering Department, Tatung Institute of Technology and NTU, from 1981 to 1983 and 1995 to 1998, respectively. Presently, he is the Dean of Electrical Engineering and Computer Science College and the Professor of Electrical Engineering Department, NTU. His research interests include digital signal processing, image processing, optical information processing, and laser holography.

Dr. Pei is a member of Eta Kappa Nu and the Optical Society of America. He received the National Sun Yet-Sen Academic Achievement Award in Engineering in 1984, the Distinguished Research Award from the National Science Council from 1990 to 1998, the outstanding Electrical Engineering Professor Award from the Chinese Institute of Electrical Engineering in 1998, the Academic Achievement Award in Engineering from the Ministry of Education in 1998, the Pan Wen-Yuan Distinguished Research Award in 2002, and the National Chair Professor Award from the Ministry of Education in 2002. He was President of the Chinese Image Processing and Pattern Recognition Society in Taiwan from 1996 to 1998. He became an IEEE Fellow in 2000 for contributions to the development of digital eigenfilter design, color image coding and signal compression, and electrical engineering education in Taiwan.