A JOINT DESIGN OF DICTIONARY APPROXIMATION AND MAXIMUM ATOM EXTRACTION FOR FAST MATCHING PURSUIT

Jian-Liang Lin†*, *Wen-Liang Hwang*† *and Soo-Chang Pei**

*Graduate Institute of Communication Engineering, National Taiwan University, Taiwan, R.O.C.
†Institute of Information Science, Academia Sinica, Taipei, Taiwan, R.O.C.

ABSTRACT

We propose a new systematic approach to reduce matching pursuit (MP) encoder complexity. MP codecs are asymmetric as decoder complexity is low while the encoder complexity is extremely high. An MP encoder contains three components: the inner products, maximum atom extraction, and atom encoding. We propose a new approach which combines the first two components using eigen-dictionary approximation and tree-based vector quantization (VQ). The advantages of this are a simpler design and a slower growth of computational costs as the target dictionary becomes large than traditional approaches. By varying the approximation accuracy, our algorithm can provide the trade-off between coding performance and speed-up of the MP encoder.

1. INTRODUCTION

Efficiently encoding motion residuals is essential for low-delay video applications in which videos are encoded by the hybrid motion compensation and a residual encoding structure. Other than nonredundant transformation, a frame based technique, matching pursuit (MP), encodes motion residual images. In [3], Mallat and Zhang first propose matching pursuit, which decomposes a signal into a linear combination of bases within an overcomplete dictionary , i.e.

$$\tilde{f}_M = \sum_{k=0}^{M-1} < R^k f, b_{j_k} > b_{j_k}.$$

The dictionary element b_{j_k} combined with the inner product value $< R^k f, b_{j_k} >$ is called an *atom*. In [4], Neff and Zakhor show that using an MP to code motion residual images performs better than using discrete cosine transform (DCT) in terms of PSNR and perceptual quality at very low bit rates. The efficacy of an MP codec depends on dictionary selection. A dictionary should meet the following three criteria: 1) the basis function must effectively represent motion residual frames, 2) the dictionary size must be small, and 3) the basis function must be simple. Because an MP encoder uses an iterative algorithm, and each iteration takes many inner product calculations, it's computational cost is higher than transform-based methods. A quick implementation of the inner products is essential for a low delay application.

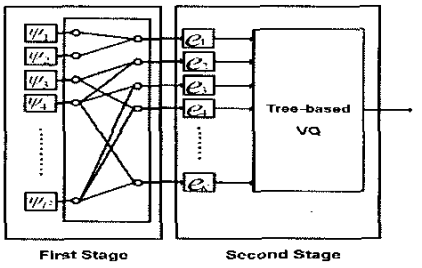

Fig. 1. The proposed two-stage-VQ structure.

A common approach to lessen inner product complexity uses a low computational cost dictionary to approximate the target dictionary. For example, a separable dictionary can be approximated by low cost factorized separable dictionary in which a large basis function is represented as a successive convolution of short basis functions [7, 2]. Another approach to lessen inner product complexity projects an arbitrary basis function into the space spanned by simple elementary functions and approximates the basis by a linear combination of the elementary functions [6, 5]. In this paper, we propose a new structure that combines a two-stage MP approach and VQ structure to efficiently approximate any dictionary. The complete structure of our two stage algorithm is given in Figure 1. The first stage is the DWT and the second stage is a design that combines the inner products of an MP residual and eigenfunctions and a tree-based VQ to extract maximum atom. The efficiency and complexity are a trade-off depended on the number of eigenfunctions K and the number of wavelet coefficients N to approximate an eigenfunction.

2. TWO-STAGE-VQ DESIGN

2.1. Dictionary approximation

At the first stage of our approach (see Figure 1), we approximate the dictionary functions by their eigenfunctions, and

0-7803-8554-3/04/$20.00 ©2004 IEEE.

each eigenfunction is then approximated by a DWT. Let the bases in the dictionary $\mathcal{D}$ be B_1,B_2, ..., $B_{|\mathcal{D}|}$. We apply PCA on the bases and select the K eigenfunctions with the largest eigenvalues. Let the K eigenfunctions be denoted as $\mathbf{E}_1$,$\mathbf{E}_2$, ..., $\mathbf{E}_K$. The eigenfunctions $\{\mathbf{E}_j\}$ are wavelet transformed using Haar wavelets denoted as $\{\psi_{mn}\}$. The Haar wavelet is used because its filtering operations can be efficiently implemented. To further reduce computational cost, each eigenfunction is approximated by the N largest DWT coefficients , i.e.

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

Ψ_i is the set of the index of the N DWT coefficients of the i-th eigenfunction.

Because $\{\mathbf{E}_i^w\}$ is an approximation of $\{\mathbf{E}_j\}$, the orthogonal property of $\{\mathbf{E}_j\}$ does not hold for $\{\mathbf{E}_i^w\}$. Using the *Gram-schmidt* procedure on $\{\mathbf{E}_i^w\}$, we have K orthonormal functions,$\mathbf{E}_j^N, j = 1, 2, ..., K$,

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

Using $\{\mathbf{E}_j^N\}$to approximate the bases $\{B_b\}$, We have

$$\hat{B}_b^N = \sum_{j=1}^{K} \alpha_{b,j}^N \mathbf{E}_j^N, \quad (2)$$

where $\alpha_{b,j}^N$ is the projection of basis B_b onto $\mathbf{E}_j^N$. An MP residual image f is approximated by $\{\mathbf{E}_j^N\}$ as

$$\begin{aligned}\hat{f}^N &= \sum_{j=1}^{K} < f, \mathbf{E}_j^N > \mathbf{E}_j^N \\ &= \sum_{j=1}^{K}(\sum_{i=1}^{j} a_{i,j} < f, \mathbf{E}_i^w >)\mathbf{E}_j^N. \quad (3)\end{aligned}$$

Using the results of Equations (2) and (3), the inner product between $\hat{f}^N$ and the normalized basis $\frac{\hat{B}_b^N}{\|\hat{B}_b^N\|}$ can be expressed as

$$\begin{aligned}&< \hat{f}^N, \frac{\hat{B}_b^N}{\| \hat{B}_b^N \|} > \\ =\ & \frac{1}{\| \hat{B}_b^N \|} \times \sum_{j=1}^{K} \alpha_{b,j}^N (\sum_{i=1}^{j} a_{i,j} < f, \mathbf{E}_i^w >) \\ =\ & \frac{1}{\| \hat{B}_b^N \|} \sum_{i=1}^{K}(\sum_{j=i}^{K} \alpha_{b,j}^N \times a_{i,j}) < f, \mathbf{E}_i^w > \\ =\ & \sum_{i=1}^{K} \alpha_{b,j}^{'} < f, \mathbf{E}_i^w >= \bar{\alpha}_b^T \bar{f}, \quad (4)\end{aligned}$$

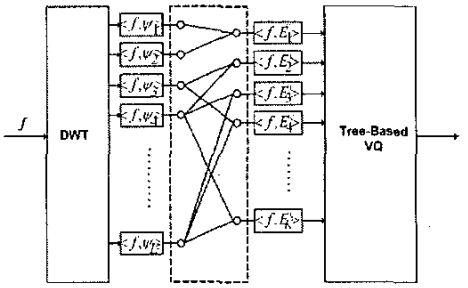

Fig. 2. A block diagram of the implementation of our proposed structure.

where

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

Hence, the inner product $< \hat{f}^N, \frac{\hat{B}_b^N}{\|\hat{B}_b^N\|} >$ can be obtained from the inner product of two K dimensional vectors.Note that $\bar{\alpha}_b$ can be pre-computed and, according to Equation (1),

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

where $\beta_{i,mn}$ can also be precomputed. The implementation of the inner product of an MP residual f and $\{\mathbf{E}_i^w\}$ is shown in Figure 2.

2.2. Tree based-VQ on an eigen-dictionary

In the traditional approach, the inner products and maximum atom extraction are two separate components. The inner products are computed first, and, after all are computed, the maximum atom is selected. In our structure, shown in Figure 2, maximum atom extraction is combined with the inner products of an MP residual and eigenfunctions.

2.2.1. Building a binary tree-VQ

There are many methods of designing a binary tree-VQ. In our MP encoder, the codewords ($\bar{\alpha}_b$) are already known, so our aim is to organize them in such a way that the binary search algorithm can find the basis whose inner product is close to what was obtained in an exhaustive search. We therefore use a simple *bottom-up* algorithm to build our binary tree-VQ. Let d be the lowest level of our tree, $d = \log_2 |\mathcal{D}|$. We use $\bar{\alpha}_b^d (= \bar{\alpha}_b)$ to assure that the codeword $\bar{\alpha}_b$ is at level d. To build the parent level, we find the pair that the gives the maximum inner product value

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

If the inner product $< \bar{\alpha}_p^d, \bar{\alpha}_q^d >$ is positive, we use the mean vector of $\bar{\alpha}_p^d$ and $\bar{\alpha}_q^d$ to represent their parent node $\bar{\alpha}_1^{d-1}$, i.e.

$$\bar{\alpha}_1^{d-1} = (\bar{\alpha}_p^d + \bar{\alpha}_q^d)/2,$$

otherwise the parent node is represented as

$$\bar{\alpha}_1^{d-1} = (\bar{\alpha}_p^d - \bar{\alpha}_q^d)/2.$$

By this same procedure, we select a pair from the remaining vectors in $\{\bar{\alpha}_b^d | b = 1, 2, \cdots, |\mathcal{D}|\} - \{\bar{\alpha}_p^d, \bar{\alpha}_q^d\}$ and construct their parent node $\bar{\alpha}_2^{d-1}$. We continue the procedure until all the vectors in $\{\bar{\alpha}_b^d | b = 1, 2, \cdots, |\mathcal{D}|\}$ are selected and the upper level $d-1$ is set up. By repeating the above procedure, we are able to build the $d-k$ level from the nodes in level $d-k+1$ until the root node in reached.

2.2.2. Querying the tree-VQ

To find which inner product of $\bar{\alpha}_b$ and an information vector $\bar{f}$ gives the maximum absolute value, 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

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

This procedure is repeated at each encountered internal node until a leaf of the tree is reached. The traditional method to extract the maximum atom (the basis that has the largest absolute inner product value) costs $\mathcal{O}(K|\mathcal{D}|)$ while our method costs $\mathcal{O}(K \log_2 |\mathcal{D}|)$. This binary tree search procedure does not always find the maximum atom; however, our experiments show that the probability of finding a basis with an inner product close to that of the optimal basis is high. The effectiveness of our binary tree-VQ will be demonstrated in Section 4.

3. COMPUTATIONAL COMPLEXITY ANALYSIS

Finding an atom from within an entire image is very time-consuming. Let us assume that the size of our basis function is L by L pixels. We use the popular suboptimal algorithm in [4, 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 energy.

A. Dyadic wavelet transform

The first step shown in Figure 2 computes the DWT of the MP residual image block, where the L by L block is centered at each pixel in the S by S local search region. All the DWT coefficients can be obtained by the *à trous* algorithm with lesser computational cost. The complexity required to implement the DWT by undecimated Haar filter bank is $6(L+S)^2 \log_2 L$ $(adds)$.

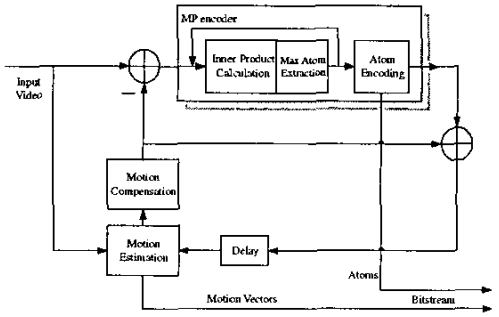

Fig. 3. The structure of an MP encoder.

B. Inner product of an MP residual and eigen-functions

The middle dashed box in Figure 2 computes the inner product of an MP residual and eigen-functions. The weights in the box $\{\beta_{i,mn}\}$ were pre-computed according to Equation (1). Because the number of dyadic wavelet coefficients to approximate each eigenfunction is N, and the number of eigenfunctions is K, the total complexity of calculating the inner products of K eigenfunctions in the local search area S^2 is $K \times N \times S^2$ $(adds + mults)$.

C. Tree search

The last step in Figure 2 applies a binary tree search to find the basis that yields a large inner product value.The complexity of finding the basis in the search block is $2 \times \log_2 |\mathcal{D}| \times K \times S^2$ $(adds + mults)$.

4. PERFORMANCE EVALUATION

We evaluate the coding performance of our two-stage-VQ algorithm using the hybrid video coding system shown in Figure 3. The first frame of a video sequence is an intra-frame (I-frame) encoded by DCT and all other frames are inter-frames (P-frames). Eleven MPEG 4 test sequences, Akiyo, Claire, Hall Monitor, Mother and Daughter, News, Salesman, Sean, Carphone, Coastguard, Container, and Miss America, are used. The size of the sequences are in QCIF format and the testing frame rate is ten frames per second. The efficiency of the proposed method depends on two parameters: the number of eigenfunctions K, and the number of dyadic wavelet coefficients N. By varying the parameters, the computational complexity and PSNR of our method will vary. Although our approach can approximate any MP dictionary, we will use the most popular separable Gabor dictionary [4] as our target dictionary. The dictionary contains 400 bases and the size of each basis is 32 by 32 pixels.

A. Y-PSNR performances

The Y-PSNR performances of our two-stage-VQ algorithm are evaluated at 17 and 30 Kbps. We use these bit rates to make representative examples to illustrate the per-

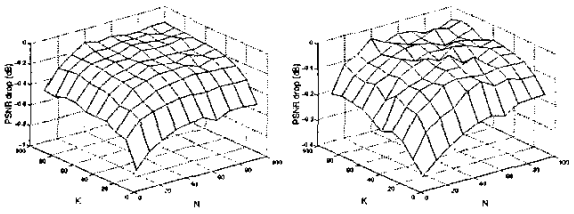

Fig. 4. The average Y-PSNR drop of the eleven sequences. Figures on the left are 30 Kbps, and figures on the right are 17 Kbps.

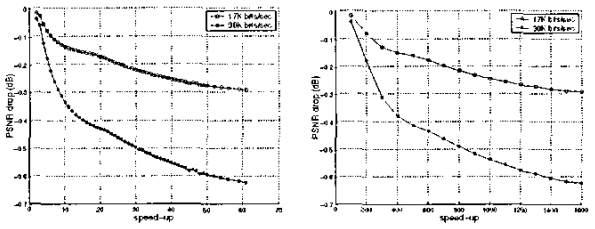

Fig. 5. Y-PSNR drop versus speed-up factor at different bit rates. Left: separable, and Right: non-separable inner product implementations of Gabor dictionary.

formance of our method. The Y-PSNR performance is measured by a Y-PSNR drop, defined as Y-PSNR of the original dictionary − Y-PSNR of the approximated dictionary. For a slow motion video sequence, such as Akiyo and Claire, the Gabor dictionary can effectively represent their motion residuals. Our experiments also show that the loss of Y-PSNR in approximating the Gabor dictionary is higher for slow motion videos than fast motion videos. In our experiments, the range of the parameters K and N is between 10 and 100. Figure 4 shows the average Y-PSNR drop for all eleven test sequences as a function of K and N at different bit rates.

B. Computing time speed-up

We compare the computation time of our two-stage-VQ to that of a separable and a non-separable dictionary. According to the complexity analysis given in Section 3, if we use *op* to denote an addition or a multiplication, the *ops* required by our two-stage-VQ to find an MP atom is $6(L+S)^2\log_2 L+2\{K\times N\times S^2+2\times\log_2|\mathcal{D}|\times K\times S^2\}$ (ops). A two dimensional separable dictionary is obtained by a tensor-product of two one dimensional dictionaries. Let N_h and N_v be the sizes of the two one-dimensional dictionaries, and $N_v\times N_h=|\mathcal{D}|$. From [4], an efficient implementation of the inner products with a separable dictionary will take $2\sum_{v=0}^{N_v-1}\{(L+S)\times S\times L+S^2\sum_{h=0}^{N_h-1}L\}+|\mathcal{D}|\times S^2$ (ops). For a non-separable dictionary, we can calculate the complexity of finding an atom of $2L^2\times|\mathcal{D}|\times S^2+|\mathcal{D}|\times S^2(ops)$. The Y-PSNR drop versus the speed-up factors of our algorithm over separable and non-separable dictionaries are shown in Figure 5. The speed-up of our algorithm can be up to 60 times faster if a target dictionary is separable, and up to 1,600 times faster if the target dictionary is non-separable.

5. CONCLUSIONS

We propose a new structure that combines a traditional two-stage MP approach and VQ structure to efficiently approximate any dictionary. When a target dictionary is large, our approach will have less design and computational complexity than a traditional two-stage approach. The proposed approach can find the first atom of a residual frame and update the inner product values for the next iterations, or it can find all atoms. We use this structure to approximate the separable Gabor dictionary, and demonstrate the trade-off between coding performance (Y-PSNR) and coding efficiency (speed-up) of our MP encoder at very low bit rates.

6. 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] P. Czerepinski, 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.

[3] G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries", *IEEE Trans. Signal Processing*,Vol. 41, pp. 3397–3415, December 1993.

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

[5] 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.

[6] D.W. Redmill, D.R. Bull, and P. Czerepinki, "Video coding using a fast non-separable matching pursuits algorithm", *Proc. IEEE Int. Conf. Image Processing.*, pp. 769–773, 1998.

[7] 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.