ADAPTIVE DICTIONARIES FOR MATCHING PURSUIT WITH SEPARABLE DECOMPOSITION

Leszek Górecki, Marek Domański

Poznañ University of Technology, Institute of Electronics and Telecommunications
Piotrowo 3A, 60-965 Poznań, POLAND
e-mail: Leszek.Gorecki@domdata.com , domanski@et.put.poznan.pl

ABSTRACT

This paper presents a learning scheme for dictionaries of two-dimensional functions for matching pursuit applied in low-bitrate video coding. The motivation is to improve the coding performance of matching pursuit compression by adapting the structure of the dictionary functions to specific types of sequences. The proposed scheme is based on separable decomposition and vector quantization. The experiments with test video sequences prove that AVC/H.264 video encoder with the proposed variant of matching pursuit coding of interframe residual exhibits improved compression performance.

1. INTRODUCTION

In video compression, motion-compensated interframe prediction is a powerful tool that transforms the input video sequence into another sequence of prediction error or prediction residual. In standard video coding techniques, transform coding is used both for intra-coding of pictures as well as for encoding of the residual. This residual has its statistical properties quite different from those for video sequences of natural scenes. Therefore, alternative mean of residual data compression are studied. An assumption is that such techniques could be better suited for residual data representation. The review of these techniques is beyond the scope of this paper. Here, matching pursuit coding is discussed [2-4] that has been already successfully applied to very low bitrate video coding.

The idea of matching pursuit is to use a large overcomplete basis set called 'a dictionary' to ensure perfect reconstruction of the original residual image. The choice and construction of the dictionary strongly affects coding performance. Nevertheless, a fundamental problem of the hitherto matching pursuit coding techniques is the lack of feedback between an input signal and a dictionary, since this technique uses a dictionary *a priori*. This fact implicates a great need for designing of a universal dictionary.

Moreover, it is assumed that the overcomplete dictionary contains functions that are able to approximate local concentrations of energy in a very accurate way. However, in practical applications, a much smaller set of basis functions is usually adopted to speed up the matching pursuit algorithm. As a result, the representation with the assistance of the universal and static dictionary is rough and not suitable enough to express subtle parts of a signal. In most matching-pursuit-based video codecs reported in the literature, a set of separable Gabor functions is used as a dictionary. This leads to a fast implementation of the matching pursuit algorithm.

Nevertheless, the most important question is if it is possible to improve representation using a constant number of functions in a dictionary. Partial answers on this questions are included in [5] and [6]. Both solutions exploit vector quantization technique. The learning scheme proposed in this paper is based on vector quantization too. The novel element of the proposed scheme is a separable decomposition, which directly gives optimal shapes.

Section 2 briefly describes a separable decomposition. The learning scheme is presented in Section 3. Finally, Section 4 shows the results of experiments.

2. MATCHING PURSUIT WITH SEPARABLE DECOMPOSITION

Matching pursuit is a technique that is able to represent a signal using small numbers of atoms. Nevertheless, computation complexity related to finding a single atom is significant. Moreover, a usual problem of matching pursuit in video coding is the lack of feedback between motion compensated residual image and a dictionary. In order to remove the mentioned inconveniences, the novel combination of a matching pursuit and a separable decomposition is proposed. In this method, an image is decomposed into two 1-D signals using separable decomposition. Then the obtained 1-D functions are approximated using one-dimensional matching pursuit algorithm. The key element of this technique is a separable decomposition.

In order to study matching pursuit with separable decomposition, let us consider a space $\aleph$ of real-value functions:

$\aleph=\{f: X\times Y\rightarrow\Re \,;\, X=[0,1,...M\text{-}1],\ Y=[0,1,...,N\text{-}1]\}$.

In this space $\aleph$, a measure $s(a,b)$ of similarity of a and b is defined :

$$s(a,b)=\frac{\left|\langle a,b\rangle\right|}{\|a\|\|b\|} \qquad a,b\in\aleph, \tag{1}$$

where $\langle\cdot\rangle$ and $\|\cdot\|$ denote the inner product and Euclidean norm, respectively.

The subset containing separable functions is denoted as $\aleph_s \subset \aleph$.

In the ideal case, a separable function $r \in \aleph$ is searched such that for each $f \in \aleph$, there is

$$\forall q \in \aleph_s \; s(f,r) \geq s(f,q). \tag{2}$$

Next, we are going to show how one can iteratively approximate the function r.

Let α_1: $Y \rightarrow \Re$, β_1: $X \rightarrow \Re$ form a separable function $q_1 \in \aleph_s$ i.e.

$$q_1(i,j) = \alpha_1(j)\beta_1(i). \tag{3}$$

Using q_1 , we can get a new separable function q_2

$$\alpha_2(j) = \sum_{i}^{M-1} f(i,j)\beta_1(i), \; j \in Y, \tag{4}$$

$$\beta_2(i) = \sum_{j=0}^{N-1} f(i,j)\alpha_2(j), \; i \in X, \tag{5}$$

$$q_2(i,j) = \alpha_2(j)\beta_2(i). \tag{6}$$

There is

$$s(f,q_2) \geq s(f,q_1). \tag{7}$$

Proof:
Let $f \in \aleph$, α_1: $Y \rightarrow \Re$, β: $X \rightarrow \Re$, $\|\alpha_1\|=1$, $\|\beta\|=1$, $q_1 \in \aleph$s, and $q_1(i,j)=\alpha_1(j)\beta(i)$.
Let assume for $\|\alpha_2\|=1/A_2$, and

$$t_2(i,j) = \alpha_2(j)\beta(i).$$

$$\langle f,t_2 \rangle = \sum_j \alpha_2(j) \sum_i f(i,j)\beta(i) = \|\alpha_2\| = \frac{1}{A_2}$$

$$\langle f,q_1 \rangle = \sum_j \alpha_1(j) \sum_i f(i,j)\beta(i) = \sum_j \alpha_1(j)\alpha_2(j)$$

Because of $\langle \alpha_1 \alpha_2 \rangle \leq \|\alpha_1\| \|\alpha_2\|$,

there is $\langle f,t_2 \rangle \geq \langle f,q_1 \rangle$.

Similarly $\langle f,q_2 \rangle \geq \langle f,t_2 \rangle$.

Waiving the assumption for $\|\alpha_1\|=1$, $\|\beta\|=1$ would leave the main idea of the proof unchanged.

Q.E.D.

It means, that a sequence of functions q_1, q_2, q_3,… approximates the signal f. In this sequence, the last function is the best separable decomposition of f.

3. LEARNING SCHEME

The separable decomposition can be used in matching pursuit algorithm to speed up the process of finding atoms. Let's remember, that the complexity of the L-approximation problem has been reduced by the greedy matching pursuit algorithm in such a way that the L dictionary elements are chosen individually instead of L at once. In an environment of separable functions, it is possible to apply a technique that dimensionally reduces the complexity of the problem in a very similar way. Note that separable decomposition finds a separable function that approximates an input signal in the best manner. This fact allows it to consider N one-dimensional functions instead of one N-dimensional signal, since a separable function can be treated as a tensor product of one-dimensional functions.

On the other hand, the separable decomposition not only reduces computational complexity of matching pursuit, but also gives a feedback to the dictionary. Note that separable decomposition computes 1-D functions and expects these functions in a dictionary. Week representation of 1-D functions causes week representation of 2-D input function. Nevertheless, the fact that optimal 1-D functions are known lays the foundation of the proposed learning scheme.

The novel learning scheme uses separable decomposition and vector quantization to result an improved dictionary. Whole process is performed as follows.

At first, an initial dictionary is used to encode the chosen sequence using matching pursuit with a separable decomposition. As a result, the set of „expected" functions is obtained. Subsequently, the vector quantization algorithm is performed on this set of functions in order to compute a next version of dictionary. The process can be repeated for the whole sequence with the assistance of the new set of functions.

The initial codebook for vector quantization (VQ) [8] is the same as the dictionary used to encode a sequence and to get training vectors $\{t_1,...,t_N\}$. Note, that at first iteration of VQ, each training vector t_j already belongs to some cell of Voronoi V_i that is represented by a dictionary function. It implies that all training vectors that were approximated by the *i*-th function in matching pursuit algorithm are classified to the same cell of Voronoi. As a result, all training vectors from a cell of Voronoi V_i define a new centroid being a new version of the *i*-th function in a dictionary.

The process of calculation of a centroid should be slightly modified to get proper results for the distance measure defined as:

$$d(a,b) = 1 - s(a,b), \tag{8}$$

where *s(a,b)* was already defined in (1).

Since *s(a,b)* depends on the absolute value of the inner product, the proper sign of the product should be considered during calculation of a centroid. This problem can be easily solved. There exists a coefficient $c_j \in \{-1,1\}$ that yields positive value of inner product:

$$\forall t_j \in V_i \quad \langle v_i, c_j t_j \rangle > 0 . \tag{9}$$

In this way, a temporary centroid is calculated as follows:

$$v_i' = \sum_{t_j \in Vi} c_j t_j . \tag{10}$$

The final centroid is obtained from v_i' by normalization i.e.:

$$v_i = \frac{v_i'}{\|v_i'\|} \tag{11}$$

The above scheme can be modified by use of expansion coefficients. In this way, the importance of training vectors may be taken into account. The experiments show that application of the weighted sum in expression (10) is not necessary as it leads to similar coding performance.

4. EXPERIMENTAL RESULTS

The purpose of the experiments was to compare an efficiency of universal dictionary proposed by Neff and Zakhor and the trained dictionaries as proposed in this paper.

Adaptation of the matching pursuit for video coding has been presented [2] and extended [3,4] by Neff and Zakhor. In their scheme, the residual of motion-compensated prediction is coded using the matching pursuit algorithm (Fig. 1) instead of classic transform coding.

In order to provide reliable results, the experimental codecs were built on top of the AVC/H.264 codec [7]. The framework of AVC/H.264 provides advanced adaptive motion-compensated prediction that ensures high coding efficiency. Application of this platform provides the results for the contemporary coding environment with matching pursuit that replace transform coding for non-intra macroblocks. The software version JM 8.4 was used for implementation of both the reference Neff and Zakhor codec as well as the authors' codec with the modified matching pursuit.

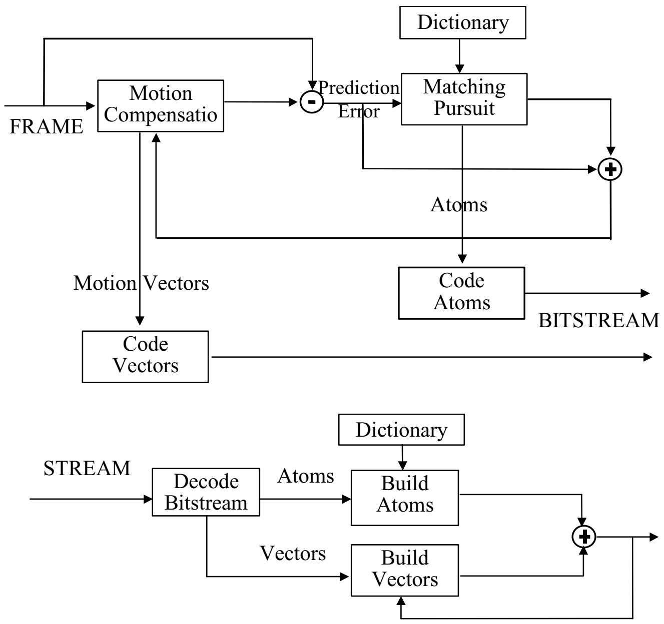

Fig. 1: The block diagram of the encoder and the decoder.

The procedure of finding atoms in matching pursuit algorithm was changed. The separable decomposition was used not only to speed-up the process of encoding but first of all to obtain the optimal shape of 1-D functions.

In the briefly described implementation, some simplifications have been made. Firstly, the atom parameters were not encoded, but instead the number of bits required to encode an atom was estimated using a statistical model based on entropy calculations. Secondly, each frame was encoded by using the number of bits known from a respective AVC bitstream, i.e. the same bit allocation as in standard AVC coding was used for the consecutive frames encoded by matching pursuit. The above simplifications are well motivated. The entropy model gives similar results as the model implemented in [2]. The synchronisation of streams gives a very good comparison model and simultaneously simplifies the control block in the experimental encoder.

Table 1. contains the rate-distortion data for matching pursuit video encoder using the dictionary proposed by Neff & Zakhor. It must be mention that this dictionary contains 20 1-D shapes (i.e. 400 separable functions).

Actually, the data are gathered for an AVC/H.264 codec with matching pursuit coding that replaces transform coding for compression of the residual of interframe motion-compensated prediction. This codec is used as a reference for assessment of the coding performance of the new tools proposed.

In the experiments, four standard QCIF (352×288) video test sequences were used: *Akiyo, Container, Silent and Foreman*. The experiments were performed for very low bitrates of about 8-48 kbps. For all test sequences, 10 seconds of video were compressed. The first frame was an I-frame and all the consecutive frames were P-frames. No B-frame was used in the experiments.

Table1. Coding performance for Neff & Zahhor dictionary .

Test sequence	Framerate [Hz]	Bitrate [kbps]	PSNR for luminance [dB]
Akiyo	7.5	8.2	34.71
Container	7.5	12.7	32.49
Silent	7.5	24.2	32.50
Foreman	10	47.8	32.78

In the first experiment, we used the proposed learning scheme to obtain optimal static and separable dictionary individually for each sequence. For this purpose, the Neff & Zakhor dictionary was used as the initial dictionary. Then, we have run several cycles of our learning scheme and have found no noticeable difference in performance between successive cycles. The obtained dictionaries give about 0.25dB increase on objective quality criteria (PSNR for luminance) (see Table 2.) to the original dictionary.

Table2. The results for individually trained Neff & Zakhor dictionary.

Sequence	Framerate [Hz]	Bitrate [kbps]	PSNR for luminance [dB]
Akiyo	7.5	8.2	34.90
Container	7.5	12.7	32.98
Silent	7.5	24.2	32.74
Foreman	10	47.8	33.01

Next experiments used randomly generated dictionary as an initial dictionary to the learning scheme. All generated dictionaries contained 20 one-dimensional shapes (similarly as Neff & Zakhor dictionary). The dictionaries were generated in the following way. At first, the region of support for each generated shape was randomly selected from the range 1 to 22. Then, the appropriate numbers of non-zero coefficients were generated. Finally, the generated function was normalized.

For each sequence, 12 randomly generated dictionaries were created. Then, each initial dictionary was used in learning scheme to obtain the optimal dictionary for each sequence. The average results (with standard deviation not greater than 0.03dB) are marked using bold style in Table 3. Next, the trained dictionaries were used to encode remaining sequences and the average results are shown also in Table 3 (normal style of font).

The results show that the proposed learning scheme is very stabile and gives similar results for any randomly generated dictionary (see diagonal in Table 3.). It is worth mentioning that results obtained form Neff & Zakhor dictionary are also similar (see Table 2.) since this dictionary can be treated as the instance of randomly generated dictionary.

Table 3. The values of PSNR [dB] of decoded luminance averaged for all sequences and for all learning schemes.

Sequence	Dictionary			
	Trained using Akiyo	Trained using Contain.	Trained using Silent	Trained using Foreman
Akiyo	**34.90**	34.77	34.85	34.90
Container	32.80	**33.01**	32.76	32.82
Silent	32.67	32.49	**32.79**	32.74
Foreman	32.98	32.82	33.02	**33.07**

As can be seen, dictionaries trained on sequences Akiyo, Silent and Foreman give similar results. This means that the above sequences contain similar characteristic of a prediction error and simultaneously its optimal dictionaries contain similar shapes. We can assume that Akiyo, Silent and Foreman belong to the same group of sequences. This mean, that dictionary that gives good results for all sequences within one type of sequences can be obtain from learning scheme performed on any representative using any initial dictionary. Therefore, it is possible to use dictionaries calculated to certain classes of video sequences. Content-class-adaptaed dictionaries may provide compression efficiency being slightly higher than that obtained with a dictionary that is not adapted to the class of video sequence content.

5. CONCLUSIONS

In this paper, an original concept of matching pursuit using separable decomposition was described, and a new learning scheme for video coding based on matching pursuit with separable decomposition has been proposed. The novel scheme has been obtained using information from a separable decomposition, which gives a feedback to the dictionary. As can be seen, further improvement of objective quality can be obtained by adaptation of dictionary to every single frame. Small improvement (e.g., about 0.25dB) may be achieved by designing dictionaries for different classes of video content like landscapes, head and shoulders, etc. The results have been verified by a series of experiments with standard test video sequences and original software that implements matching pursuit coders on the platform of advanced motion-compensated prediction of AVC/H.264.

REFERENCES

[1] S. Mallat, Z. Zhang, *Matching Pursuits With Time-Frequency Dictionarie*s, IEEE Transactions in Signal Processing, Vol. 41, No. 12, pp. 3397-3415, December 1993.

[2] R. Neff, A. Zakhor, *Matching Pursuits Video Coding as Very Low Bit Rates*, IEEE Data Compression Conference, Snowbird, Utah, pp. 411-420, March 1995.

[3] R. Neff, A. Zakhor, *Very Low Bit Rate Video Coding Based on Matching Pursuit*, IEEE Transactions Circuits and Systems for Video Technology, Vol. 7, pp. 158-171, February 1997.

[4] R. Neff, A. Zakhor, *Matching Pursuit Video Coding – part I: Dictionary Approximation*, IEEE Transactions Circuits and Systems for Video Technology, Vol. 12, pp. 13-26, January 2002.

[5] P. Schmid-Saugeon, A. Zakhor, *Learning Dictionaries for Matching Persuits Based Video Coders*, IEEE ICIP, Vol. 3, pp. 562-565, Thessaloniki 2001.

[6] Chien-Kai Peng, Chung-Lin Huang, Wen-Liang Huang, *Very-Low-Bit Rate Coding using Matching Pursuit and Codebook Adaptation*, ICIP, 2000.

[7] ISO/IEC JTC1/SC29/WG11, Doc. N6540, Text of ISO/IEC 14496-10 Advanced Video Coding, 3-rd Edition, July 2004.

[8] A. Gersho, R. Gray, *Vector quantization and signal processing*, Kluwer, Boston 1992.