MOTION ESTIMATION FOR VIDEO CODING BASED ON SPARSE REPRESENTATION

Yanfei Shen[1,2], Jintao Li[1], Zhenmin Zhu[1]
1 Institute of Computing Technology, Chinese Academy of Sciences, Beijing, P.R. China
2 University of Chinese Academy of Sciences, Beijing, P.R. China
{syf, jtli, zmzhu}@ict.ac.cn

ABSTRACT

This paper describes a motion estimation algorithm based on sparse representation, which can be applied in video coding to reduce the temporal redundancy. The sparse coefficients are firstly calculated in support region by orthogonal matching pursuit (OMP) algorithm using the reference blocks as dictionary elements, and then these optimal sparse coefficients are utilized to predict the current block. To get the same prediction in decoder, the number of iterations in OMP is transmitted to decoder as side information. Simulation results show that gain up to 2.87dB in terms of the PSNR when compared with traditional translational motion estimation model.

Index Terms—motion estimation, sparse representation, video coding

1. INTRODUCTION

Motion estimation (ME) is the essential element in video coding which is used to reduce to temporal redundancy. The performance of ME algorithms has a great influence on the video coding efficiency. Therefore, in the last few decades, much research work is done to improve the prediction accuracy of ME algorithms, including variable block size, complex affine motion model, fractional motion vector precision and sub-pixel interpolation filter etc.

However, there are two unsolved key problems which affect the video coding efficiency. The first is motion estimation accuracy problem. In most of the popular video coders, such as H.264[1], HEVC[2] etc., motion estimation is based on rectangular block, called macro block, which is compared with the corresponding block and its adjacent neighbors in the reference frames to create a motion vector that represents the movement of objects. These motion estimation algorithms assume that the motion in video sequences is rigid and generally a pure translation motion model is used, but objects in the real world undergo more complicated motions, it is difficult for a pure translation motion model to adapt the complex motion of nature video sequences. Although various efforts have been made to use more complex motion models for motion compensated prediction, for example, one of the early proposals for the H.264/AVC standard was a codec based on an affine motion model[3][4]. However, it is difficult to estimation the affine motion parameters from the reconstructed video sequences. The second is the coding overhead for motion parameter. For pure translation motion model, smaller sub-block can improve motion estimation precision, such as, in H.264/AVC video coding standard, the smallest sub-block is 4x4, but the side information of motion vectors is increased correspondingly. In addition, although affine motion model can provide better motion representation than translation motion model, its disadvantage is the increased number of motion vector parameters and motion vector bit rate, the encoded motion vectors consume up to 25% of the total frame bit rate [5].

In the past decade, sparse representation based algorithms have sparked a great research interest for signal processing and image compression with numerous applications, e.g. image prediction [6], image denoising [7], restoration [8], compression [9], and more. The main idea behind sparse representation is that a signal $y \in R^n$ can be represented as a linear combination of few prototype signals from a dictionary $D \in R^{n \times m}$, which contains a collection of m atoms $d_i \in R^n$ that are building blocks of the representation. A prediction algorithm based on sparse representation has been introduced in [10], in this method, the basic functions which best approximate a causal neighborhood are used to extrapolate the signal in the region to predict. An online dictionary learning method is proposed to address the problem of intra image prediction based on signal expansion on overcomplete dictionaries [6]. Two spatial image prediction methods based on nonnegative matrix factorization (NMF) and locally linear embedding (LLE) have been introduced in [11], these two methods approximate the current block as a linear combination of k-*nearest* neighborhood of the input block. All of these algorithms in various image processing assume that the natural images are composed of only a few structural primitives. One has thus to first learn these primitives and then decompose the image on the set of primitives to extract the representative features of the image.

In this paper, we will propose a ME algorithm based on sparse representation to reduce the time redundancy in video coding. If we imagine the reference blocks as dictionary

atoms and the current predict block as prediction vector, the block matching process is really sparse representation process where the sparse coefficient is one for full pixel motion estimation. In our proposed method, motivated by sparse approximation techniques introduced in [10], the neighborhood known pixels of the current block are firstly sparse coding by OMP method [12], then the same sparse coefficients are used to predict the current block. The number of sparse coefficients decides the used motion models. So the proposed method can adapt the complex motion and texture features of natural video sequences. In addition, because the sparse coefficients are calculated by known pixels, decoder can repeat the identical operations. It is not necessary to transmit side information of sparse coefficients.

The rest of this paper is organized as follows. In section 2, we will recall the algorithm for sparse representation by OMP, and its relationships with traditional ME method. Our proposed ME algorithm based on sparse representation will be described in section 3. Section 4 gives the experimental results in terms of prediction quality and some simple analysis. Finally, we conclude this paper in section 5.

2. SPARSE REPRESENTATION AND MOTION ESTIMATION

The basic model of sparse representation suggests that the nature signals can be efficiently explained as linear combinations on an overcomplete dictionary, where the linear coefficients are sparse (most of them are zeros). Formally, if x is a column signal and D is the dictionary (whose columns are the atom signals), the sparse representation can be described by the following sparse approximation problem,

$$\alpha = Arg\min_{\alpha} \|x - D\alpha\|_2^2 \quad s.t. \quad \|\alpha\|_0 \le k \tag{1}$$

In this formulation, α is the sparse representation coefficient of signal x, k is sparse degree of α, which is the ℓ^0 pseudo-norm counting the non-zero entries. The solution to this approximation problem can be efficiently solved using several available approximation techniques, including OMP, Basis Pursuit, FOCUSS, and others [13].

The traditional ME algorithm can be viewed as a special case of sparse representation, where the observed signal x of sparse representation corresponds to the current block of motion estimation and the dictionary D of sparse representation corresponds to the reference blocks in the search window of reference frame. Motion estimation process is to search one atom in dictionary which is the most relevant to the current block and the index of selected atom is equivalent to the output motion vectors. For integer pixel motion estimation, the number of sparse coefficients is one and its value is also one integer, that is to say, $\|\alpha\|_0 = 1$. For fractional pixel translational motion estimation model, the sparse coefficients are relevant to the coefficients of interpolation filter f which is used to generate the fractional pixels value and the accuracy of motion vectors as follows,

$$\alpha = Arg\min_{\alpha} \|x - Df\alpha\|_2^2 \quad s.t. \quad \|f\alpha\|_0 \le k \tag{2}$$

Where f is the coefficient of interpolation filter. For example, if bilinear interpolation $f = \{0.5, 0.5\}$ is used in half pixel motion estimation, the optimal sparse coefficients will be $\{0.5, 0.5\}$ and $\|f\alpha\|_0 = 2$. For other complex motion estimation models, including polynomial affine motion, sparse representation can be optimal solution and can be adapt to various motion and texture context.

3 MOTION ESTIMATION ALGORITHM BASED ON SPARSE REPRESENTATION

This section firstly describes the main principle of the ME algorithm based on sparse representation and then discusses how this algorithm can be applied to reduce the time redundancy in video coding.

Let C denote the current block to be predicted by motion estimation algorithm, and its causal neighborhood S used as sparse representation support, as shown in Fig.1. The basic principle of our proposed method is to first search for a good approximation of known pixels in S region and to calculate its sparse representation coefficients, and then keep the same procedure to estimate the unknown pixel value in C region.

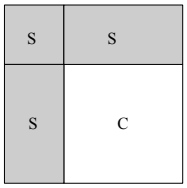

Fig.1 The current Block C and its support region S

Let the pixel value of the support region S and current block C be arrayed in column vector b_s and b_c, the vector b is finally comprised of b_s and b_c as follows,

$$b = \begin{bmatrix} b_s \\ b_c \end{bmatrix} \tag{3}$$

Let D denote sparse representation dictionary by a matrix of dimension of $N \times M$, where N is equal to the number of elements in vector b and M is decided by the size of search windows in traditional ME algorithm. The columns of dictionary D are constructed by the same method as vector b. The use of causal neighborhood S guarantees that the decoder can construct the same dictionary. The dictionary

matrix D is then assumed to be formed by two sub-matrices D_s and D_c as follows,

$$D = \begin{bmatrix} D_s \\ D_c \end{bmatrix} \tag{4}$$

The basic idea of our method is to first search for a linear combination of atoms taken from the dictionary D_s, which best approximates the pixel value in the support region S, and then to keep the same linear combination, including the same indexes of selected atoms and the same sparse coefficients, to estimate the pixel value in current block C. Sparse representation algorithms aim at solving the approximation minimization as

$$\min \|b_s - D_s\alpha\|_2^2 \quad s.t. \ \ \min \|\alpha\|_0 \tag{5}$$

Where α is sparse coefficient. In practice, one actually seeks an approximate solution which satisfied:

$$\min \|\alpha\|_0 \quad s.t. \ \ \|b_s - D_s\alpha\|_p \le \varepsilon \tag{6}$$

for some $\varepsilon \ge 0$, characterizing an admissible reconstruction error. The norm p is usually 2. This problem is known to be NP-hard and different sub-optimal strategies have to be used. There are generally based on convex relaxation of the problem, non-convex local optimization or greedy search algorithms. Greedy algorithms, including MP and OMP, have been introduced as heuristic algorithms to find approximate solutions with tractable complexity.

In this paper, the OMP algorithm will be used and it proceeds as follows. At the first iteration, $\alpha_0 = 0$ and an initial residual vector $r_0 = b_s - D_s\alpha_0$ is computed. At iteration k, in order to find a better approximation for b_s, the algorithm will determine an atom a_i^k to be selected from dictionary D_s which has maximum correlation with the previous residual vector r_{k-1}. In particular, the selection is based on the inner products between residual vector r_{k-1} and the atom d_i of dictionary D_s

$$a_i^k = \arg_i \max \left| r_{k-1}^T d_i \right|, \quad d_i \in D_s \tag{7}$$

Let D_s^k denote the compacted matrix containing all the atoms selected in previous iterations

$$D_s^k = D_s^{k-1} \cup a_i^k \tag{8}$$

The new coefficient vector and residual vector at the *kth* iteration are given as follows

$$\alpha_k = (D_s^{kT} D_s^k)^{-1} D_s^{kT} b_s \tag{9}$$

$$r_k = b_s - D_s\alpha_k \tag{10}$$

There are several natural stopping criteria for OMP, including a fixed number of iterations or a threshold of residual magnitude. However, these stopping criteria is not suitable to our proposed algorithm, because the sparse coefficients which lead to small energy of residual in support region are not necessarily accurately predict the current block. In this paper, we will use the energy of residual on the current block as the stopping criteria, that is, if it is smaller than some threshold, the OMP will stop. The number of selected atoms (also the number of iteration k) that minimize the above criterion is transmitted to the decoder as side information. The decoder similarly runs the algorithm with the same dictionary and the same support region which has been decoded and reconstructed previously, the number of selected atoms can thus be used as stopping criterion, so the motion estimation algorithm at the encoder and the motion compensation algorithm at the decoder can get the same sparse coefficients α_{opt} which can be used to predict the current block. The pixel value of current block is then calculated by multiplying the dictionary D_c by α_{opt} as. $b_c = D_c\alpha_{opt}$. The complete mathematical description of our proposed motion estimation algorithm based on sparse representation is summarized in Table I.

Table I ME Algorithm Based on Sparse Representation

Input: b_s, D_s, D_c, k

Output: Predicted values b_p, the number of iteration k^*

1) Initialization: $k = 0$, $\alpha_0 = 0$, $r_0 = b_s$, $D_s^k = \phi$

2) ME based on sparse representation

Do until $k = K$

$k = k + 1$

- Find an atom a_i^k of dictionary D_s that is most strongly correlated with the residual r_k:

  $a_i^k = \arg_i \max \left| r_k^t d_i \right|, d_i \in D_s$

  $D_s^k = D_s^{k-1} \cup a_i^k$
- Find the best coefficients for approximating the pixels in support region with the selected atoms D_s^k so far

  $a_k = \arg\min \left\| b_s - D_s^k \alpha \right\|_2$
- Calculate the energy of residual on current block by sparse coefficients a_k

  $E_k = \arg\min \left\| b_c - D_c\alpha_k \right\|_2$
- Update the residual

  $r_k = b_s - D_s^k\alpha$

end do

Select the optimum k^* that minimize the energy of residual E_k

3) **Prediction of current block**

Calculate the prediction pixel values of the current block

$b_p = D_c\alpha_{k^*}$

4) **Output**

Predicted values b_p of current block b_c, the number of iteration k^*

4 EXPERIMENTAL RESULTS

In this section, in order to evaluate the performance of our proposed method, the proposed method was tested using different standard video sequences, including foreman, coastguard, Waterfall etc with CIF resolution. The traditional translational motion model with sub-pixel motion vector accuracy is used and compared. The block size is set 8x8 pixels and the search range of motion estimation is ±16 pixels. The pixel value in sub-pixel location is generated by bilinear interpolation filter and the search method in translational motion model is full search algorithm. The reference frame used in our experiment is the original frame and the extrapolation pixel beyond the image edges is repeated by the nearest pixel. The size of the current block and support region is shown in Fig.2.

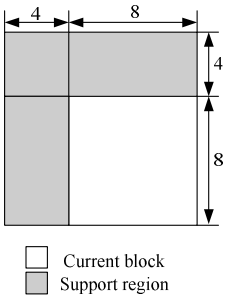

Fig.2 the size of current block sand support region

The performance of motion estimation algorithm is measured by PSNR between the current original frame and the reconstructed frame, as shown in Table II. It is demonstrated that our proposed motion estimation algorithm can outperform the traditional translational motion estimation (TME) method in most video sequences with complex texture, such as waterfall, flower and container etc; the most gain can get 2.87dB. The temporal redundancy can be efficiently reduced by the sparse representation for video coding. However, for video sequences with strong edge, such as bus and tempete, because the texture property of the support region and current block may be inconsistent, it is less efficient to use sparse coefficients trained from support region to predict the current block, the performance of our proposed motion estimation is lower than the traditional translational motion estimation method.

To guarantee the same prediction for the current block in video decoder, the number of iteration for every block is needed to be transmitted as side information. The distribution of iteration number is shown in Fig.3. If the number of iteration is one, our proposed motion estimation algorithm corresponds to traditional translational motion model with full pixel accuracy, otherwise, it corresponds to fractional pixel motion estimation algorithm, but the difference between them is that the coefficients of interpolation filters used to generate fractional pixel value in our proposed method are calculated in real time and can be adapt to real motion model hidden in video sequence. In addition, the bit overhead of transmitting this side information is lower than motion vectors. Fig.3 shows that the major of iteration numbers is less than six, that is, the current block is predicted by linear combination of less six reference block, this is coincide with the length of interpolation filter defined in video coding standard H.264.

Table II PSNR performance of TME algorithm and proposed motion estimation algorithm (dB)

sequence	TME	Proposed	Gain
Forman	36.65	36.90	**0.25**
Flower	30.75	32.66	**1.91**
Bus	27.45	26.24	-1.21
Container	38.44	41.31	**2.87**
Coastguard	33.15	33.19	**0.04**
Tempete	29.89	28.69	-1.2
Waterfall	37.46	39.57	**2.11**

In traditional translational motion model, the coefficients of interpolation filter are fixed and there are many fast motion estimation algorithms, so its computational burden is lower. The coefficients of sparse representation are calculated in real time for our proposed motion estimation algorithm, so it has more computation complexity. However, there are many fast OMP algorithms [14], such as stage wise orthogonal matching algorithm, which can be used to speed up our algorithm.

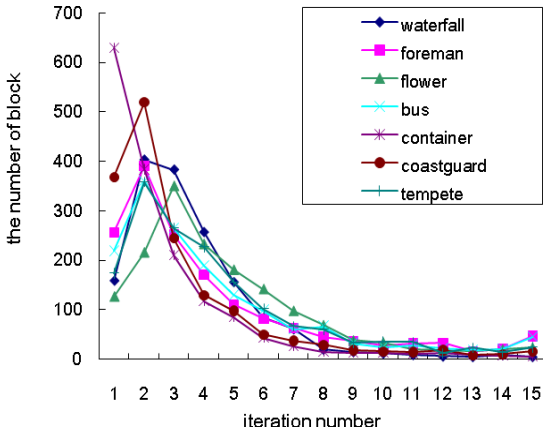

Fig.3 the distribution of iteration number

6. ACKNOWLEDGEMENT

This task is supported by the Nation Natural Science Foundation of China (Grant No.61001123)

REFERENCES

[1] T.Wiegand, G.J. Sullivan, B. Bjontegaard, and A. Luthra, "Overview of the h.264/avc video coding standard," IEEE Trans. on Circuits and Systems for Video Technology, vol. 13-7, pp. 560–576, July 2003.

[2] T. Wiegand, W.J. Han, B. Bross, and J. R Ohm, and G.J. Sullivan, "Working Draft 3 of High-Efficiency video Coding," JCTVC-E603, Geneva, CH, Mar 2011.

[3] M. Karczewicz, J. Nieweglowski, and P. Haavisto, "Video coding using motion compensation with polynomial motion vector fields," Signal Process.: Image Commun., vol. 10, pp. 63–91, 1997.

[4] MVC Coder/Decoder Submitted to ITU-T Nokia Inc.—Nokia Research Center, 2000 [Online]. Available: http://ftp3.itu.ch/av-arch/video-site/

[5] R. C. Kordasiewicz, M. D. Gallant, and Shahram Shirani, "Encoding of Affine Motion Vectors", IEEE Trans. on Multimedia, vol.9, No.7, pp 1346-1356,Nov. 2007.

[6] Mehmet Turkan, and Christine Guillemot, "Online Dictionary for Image Prediction", IEEE Int. Conf. Image Processing, 2011, pp.293-297.

[7] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," IEEE Trans. Image Process., vol. 15-12, pp. 3736–3745, Dec. 2006.

[8] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," IEEE Trans. Image Process., vol. 17-1, pp. 53–69, Jan. 2008.

[9] O. Bryt and M. Elad, "Compression of facial images using the K-SVD algorithm," J. Visual Commun. Image Represent., vol.19-4, pp. 270–283, May 2008.

[10]A. Martin, J.-J. Fuchs, C. Guillemot, and D. Thoreau, "Sparse representation for image prediction," in Proc. Eur. Signal Process. Conf.,2007.

[11] Mehmet Türkan and Christine Guillemot, "Image Prediction Based on Neighbor-Embedding Methods", IEEE Trans. Image Process., vol. 21-4, pp. 1885–1898, Apr. 2012.

[12] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in Conf. Rec. 27th Asilomar Conf. Signals, Syst. Comput., 1993, vol. 1.

[13] By Joel A. Tropp and S.J. Wright, "Computational methods for sparse solution of linear inverse problems", Proceedings of the IEEE, vol.98-6, pp 948- 958, jun.2010.

[14] B. L. Sturm and M. G. Christensen, "Comparison of Orthogonal Matching Pursuit Implementations, EUSIPCO 2012, Bucharest, Romania, Aug. 2012.