Fast Matching Pursuit Method Using Property of Symmetry and Classification for Scalable Video Coding

Soekbyeung Oh[1], Byeungwoo Jeon[2]
School of ECE, Sungkyunkwan University
300 Chunchun-dong, Jangan-gu, Suwon, Kyunggi-do, Korea
Tel : +82-331-290-7186, Fax : +82-331-290-7191
E-mail : osb@ece.skku.ac.kr[1], bjeon@yurim.skku.ac.kr[2]

ABSTRACT

Matching pursuit algorithm is a signal expansion technique whose efficiency for motion compensated residual image has already been demonstrated in the MPEG-4 framework. However, one of the practical concerns related to applying matching pursuit algorithm to real-time scalable video coding is its massive computation required for finding dictionary elements. In this respective, this paper proposes a fast algorithm, which is composed of three sub-methods. The first method utilizes the property of symmetry in 1-D dictionary element and the second uses mathematical elimination of inner product calculation in advance, and the last one uses frequency property of 2-D dictionary. Experimental results show that our algorithm needs about 30% computational load compared to the conventional fast algorithm using separable property of 2-D gabor dictionary with negligible quality degradation.

1. INTRODUCTION

The realization of real time video communication on internet-based application needs many requirements at various environments. Especially following three outstanding points are very important. First is the maximum data channel capacity, and second is the stability of communication channel. And the last is ability of real time processing of hardware. But thinking of today's technology and internet-environment, there are many users having only low bit-rate channel, and time variant channel capacity. Moreover from the point of view in hardware, there are only a few devices supporting thirty frames per second for video encoding. So for the purpose of real time video communication, those problems have to be taken into consideration. To solve the real time problems, there have been many researches [1]-[3]. To overcome the low bit rate channel capacity, matching pursuit video encoding method was introduced by Neff [1]. This method has been proven to be excellent in low bit rate video coding than conventional DCT based method.

Matching pursuit method has another useful advantage in aspects of scalable coding algorithm. It is useful when the channel between users is different or the channel capacity changes with time. Therefore matching pursuit method is apt to internet-based video communication coding even though there are some practical problems. The most important problem is computational complexity of matching pursuit. The major computational complexity comes from the fact that the inner product of the input image signal is computed with respect to all dictionary functions and all points within search size to find an dictionary element which best matches the given image function. For this method to be any useful in real applications, practical solution should be devised to speed up the encoding process. In this respect, we introduce a new method to reduce computational loads. Our method uses symmetry property of 1-D dictionary element, and classifies 2-D dictionary into four with frequency property. This paper is arranged as follows. The basic matching pursuit video coding method is reviewed in section 2. Section 3 presents our new fast method for matching pursuit. Section 4 shows experimental results, and concludes at section 5.

2. MATCHING PURSUIT

The matching pursuit encoder decomposes 2-D motion residual image $f(x,y)$ with 2-D dictionary element $g_{\alpha,\beta,n}(x,y)$ and weighting value p_n over multiple stages. Then input residual image is represented by below:

$$f(x,y)=\sum_{n=0}^{\infty} p_n \cdot g_{\alpha,\beta,n}(x,y)$$

At each stage, single basis element $g_{\alpha,\beta,n}(x,y)$ is

This work was supported in part by the Ministry of Information & Communication of Korea ("Support Project of University Foundation Research 1999" supervised by IITA).

chosen from the 2-D dictionary G in order to best approximate the updated residual image $R_n(x,y)$. Weighting value p_n is acquired from inner product calculation:

$$p_n = \left\langle g_{\alpha,\beta,n}(x,y), R_n(x,y) \right\rangle$$

After the first stage, the input residual image is decomposed as:

$$f(x,y) = p_0 \cdot g_{\alpha,\beta,0}(x,y) + R_1(x,y)$$

Here, $R_1(x,y)$ is the first updated residual image. The same process is continued until it consumes all assigned bit budget.

The most common dictionary element of matching pursuit is a general family of time-frequency basis and can be generated by scaling, translating, modulating a single 1-D function $g(x)$ with following equation form:

$$g(x) = \sqrt[4]{2} e^{-\pi t^2}$$

$$g_\alpha(x) = \frac{1}{\sqrt{s}} g\left(\frac{x-u}{s}\right) e^{i\xi}$$

Where, we imposed that $|g_\alpha(x)| = 1$, scale variable is $s > 0$, frequency modulation factor ξ, translation factor is u, and $\alpha = (s, \xi, u)$. With above 1-D dictionary, we can easily expand it to 2-dimensional dictionary.

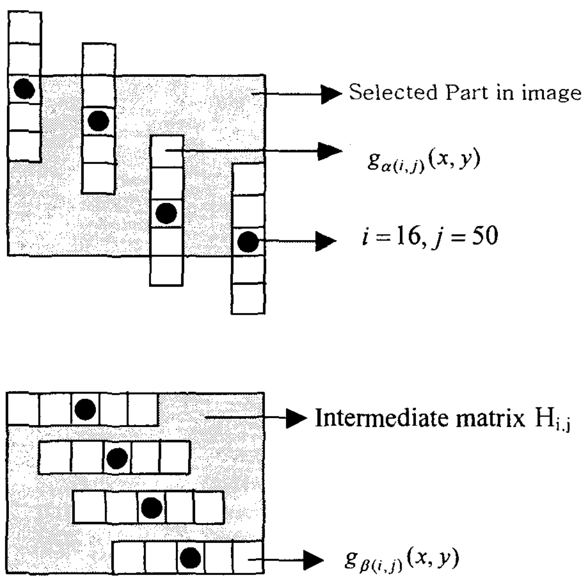
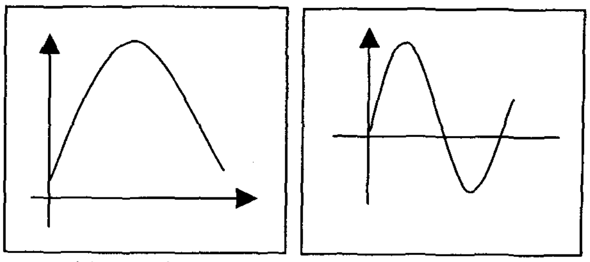


Fig. 1. Fast matching pursuit based on separability



(a) Pseudo even (b) Pseudo odd

Fig. 2. Pseudo even and odd element

3. PROPOSED ALGORITHM

In matching pursuit video coding method, we should calculate the value of inner product 2-D residual image with 2-D dictionary element within all search size. But this calculation process is divided into two sub stages. At first, we calculate the 2-D residual image with vertical 1-D dictionary element, this result makes new 2-D intermediate matrix. And at second stage, we can calculate the result of 2-D residual image with 2-D dictionary element by calculating the value of inner product horizontal 1-D dictionary with corresponding intermediate matrix element. The detailed description is shown in figure 1. We utilize two properties of Gabor dictionary and one mathematical method to speed up the matching pursuit encoding.

3.1 Property of 1-D Gabor dictionary element

Multiplying of typical Gaussian window function and cosine function generates the 1-D Gabor dictionary elements with various scaling, translating, modulating factor. Resulting 1-D dictionary elements are pseudo even or pseudo odd function as shown in figure 2. Using this property we can reduce the number of multiply-calculation about half. Let's think of two signals. One is the 1-D horizontal dictionary element discrete signal g[n] and the other is corresponding element signal h[n]. In pseudo even case,

$$g[1] = a, g[2] = b, \cdots, g[m] = c, \cdots, g[n-1] = b, g[n] = a$$
$$h[1] = h_1, h[2] = h_2, \cdots, h[m] = h_m, \cdots, h[n-1] = h_{n-1}, h[n] = h_n$$

The inner product value is calculated below reduced form.

$$\begin{aligned} & a \cdot h_1 + b \cdot h_2 + \cdots + c \cdot h_m + \cdots + b \cdot h_{n-1} + a \cdot h_n \\ &= a \cdot (h_1 + h_n) + b \cdot (h_2 + h_{n-1}) + \cdots + c \cdot (h_m) \end{aligned}$$

The pseudo odd case is the same. And all case of 1-D Gabor dictionary is same too, because symmetry property of them. The reduced computational loads converge to a half with larger element number or size [4].

3.2 Mathematical elimination method

After the first stage in the Neff's fast algorithm, we have an intermediate matrix, which component are results from values of above first stage and are used by corresponding element of 1-D horizontal dictionary when inner product calculation is operated at the second stage. Like as the first method, let think of two signals. One is the 1-D horizontal dictionary element discrete signal g[n] and the other is corresponding element signal h[n]. Two signals are discrete and of the same size, so we induce useful relationship for inner product calculation. We can suppose that each discrete signal is in below form:

$$h[1]=h_1,\ h[2]=h_2,\ \cdots\cdots,\ h[n]=h_n$$
$$g[1]=g_1,\ g[2]=g_2,\ \cdots\cdots,\ g[n]=g_n$$

So, we can represent inner product as below:

$$\langle h[n],g[n]\rangle = h_1\cdot g_1 + h_2\cdot g_2 + \cdots + h_n\cdot g_n$$

Here, because all 1-D dictionary elements are normalized, below relationship exist in all cases.

$$\begin{aligned} h_1\cdot g_1 &\le |h_1|, \\ h_2\cdot g_2 &\le |h_2|, \\ &\vdots \\ h_n\cdot g_n &\le |h_n| \end{aligned}$$

Summarizing each side gives useful relationship:

$$\langle h[n].g[n]\rangle \le \sum_{i=1}^{n}|h_i|$$

Using above relationship we can remove dictionary element that can be never an atom without calculating inner product.

3.3 Property of 2-D Gabor dictionary element

The last property that we use in fast inner product calculation is the frequency characteristic of 2-D elements of dictionary. That is to say, entire dictionary can be divided into four parts according to their dominant frequency. The four groups in Figure 3 is characterized as:

Part (a): the region that dictionary element has much DC component.
Part (b): the region that dictionary element has much horizontal frequency component.
Part (c): the region that dictionary element has much vertical frequency component.
Part (d): the region that dictionary element has high frequency component.

We can make decision what part will be used in finding atom process. Two DCT-coefficient (c(0,1), c(1,0)), and

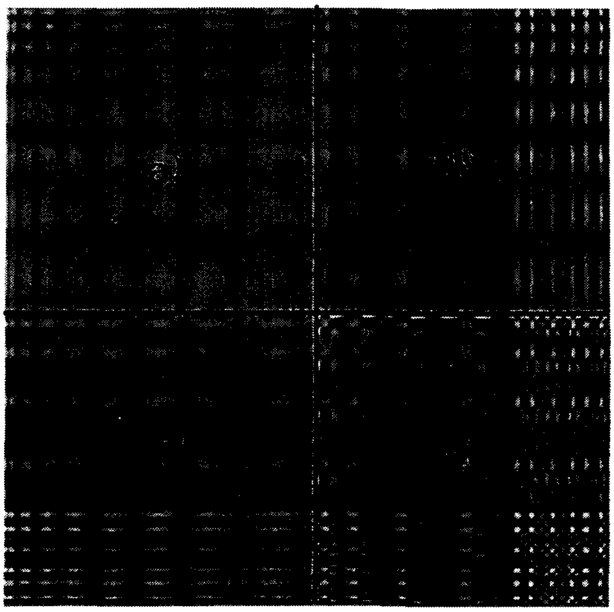


Fig. 3. Partition of Gabor dictionary

one summing value of gray level(=T) within search size is that.

$$c(u,v)=\frac{2}{N}A(u)A(v)\sum_{0}^{N-1}\sum_{0}^{N-1}f(x,y)COS(\frac{(2x+1)u\pi}{2N})COS(\frac{(2y+1)v\pi}{2N})$$

$$T=\sum_{i=0}^{S-1}\sum_{j=0}^{S-1}f(x,y)$$

Decision criterion is as below:

1. Part (b), (c) and Part (a), (d) is discriminated by ratio of $c(0,1)$ and $c(1,0)$. Because the ratio of Part (a) and Part (d) is very close to one-value.
2. Part (b) and Part (c) is discriminated with $c(0,1)$ and $c(1,0)$. Because they are representative vertical and horizontal component respectively.
3. Part (a) and Part (d) is discriminated with T, the summing value in Part (a) is very large, but that value of Part (d) is close to zero-value.

4. EXPERIMENTAL RESULTS

In our experiments, we use two layers for scalable coding. One is base-layer coding that use first and second proposed method for speed up encoding process and no image degradation. The other is enhance-layer coding that uses all three proposed method for much speed up encoding process. In this experiments we tested QCIF images, which are monochrome with 256 gray levels. Using Gabor dictionary composed of four hundred elements (Vertical is 20 (V=20), horizontal is 20 (H=20)), and search size is 16 (S=16).

4.1 First method result.

The needed amount of multiply-calculation for finding one atom in Neff's fast algorithm is about 1,700,000 which amount is calculated below:

$$M = \sum_{v=0}^{V-1} (L_V \cdot S(S + L_{\max} - 1) + S^2 \sum_{h=0}^{H-1} L_h)$$

Where L_v, L_h is length of each 1-D dictionary elements, and $L_{\max}$ is the length of element which has maximum length in 1-D Gabor dictionary element. When our proposed method is used, the amount of calculation is reduced as below:

$$M = \sum_{v=0}^{V-3} ((\frac{L_v + 1}{2}) \cdot S(S + L_{\max} - 1) + S^2 \sum_{h=0}^{H-3} (\frac{L_h + 1}{2}) + S^2 \sum_{h=H-2}^{H-1} L_h)$$
$$+ \sum_{v=V-2}^{V-1} (L_v \cdot S(S + L_{\max} - 1) + S^2 \sum_{h=0}^{H-3} (\frac{L_h + 1}{2}) + S^2 \sum_{h=H-2}^{H-1} L_h)$$

4.2 Second method result

Tested result is shown Table 1. This method reduces about 40% calculation of inner product.

Table 1. Experimental result (I)

Sequence	Number of inner product calculation for one atom		Ratio (N/P) (%)
	Neff&Zakhor (N)	Proposed (P)	
Akiyo	102400	47567	46
Foreman	102400	37779	37
Miss America	102400	44434	43
Suzie	102400	43961	42

This method includes negligible additional calculations. Those are N*S*(L_{max}-1)=1600 addition and absolute calculation per one atom, but it is less than 1% of inner product calculation.

4.3 Third method result

This method reduces the amount of multiply-calculation 75% than Neff's fast algorithm, on account of using one of four parts is used. But this method has a little image quality degradation. The reconstructed image is averagely 0.2dB lower in PSNR than Neff's fast algorithm. Table 3 shows experimental result.

Table 2. Experimental result (II)

Sequence	AVERAGE PSNR	
	Neff&Zakhor	Proposed
Akiyo	33.92	33.71
Foreman	28.28	28.02
Miss America	38.64	38.51
Suzie	32.92	32.68

4.4 Scalable coding result

In scalable coding method, atom is divided two parts. One is base-layer atom, and the other is enhance-layer atom. We find atom using first and second method in base-layer encoding for giving no image degradation. And using all three methods for more speed up in enhance-layer encoding that involves little image degradation, because it is operated on updated residual image at base-layer encoding process. Table 3 represents experimental results that we use 50 atoms in base-layer and 50 atoms in enhance-layer and thirty frames at each image.

Table 3. Experimental result (III)

Sequence	AVERAGE PSNR			
	Base-layer		Enhance-layer	
	Neff & Zakhor	Proposed	Neff & Zakhor	Proposed
Akiyo	33.92	33.92	36.69	36.58
Foreman	28.28	28.28	34.27	34.12
Miss America	38.64	38.64	38.84	38.78
Suzie	32.92	32.92	36.22	36.10

5. CONCLUSIONS

In this paper, we proposed a fast matching pursuit algorithm that composed of three methods. This algorithm reduce about 70 ~ 80% of inner product calculation without image degradation in base-layer encoding process, and can reduce about 90% than Neff's fast algorithm in enhance-layer encoding process.

REFERENCES

[1] R. Neff and A. Zakhor, "Very Low Bit Rate Video Coding Based on Matching Pursuit," IEEE Trans. Circuits and Systems for Video Technology, Vol. 7, pp. 158-171, Feb. 1997.

[2] P. Czerepinski, C. Davies, N. Canagarajah and D. Bull,"Dictionaries and Fast Implementation for Matching Pursuits Video Coding," Proceedings of the Picture Coding Symposium, pp. 41-44, 1999.

[3] M. Gharavi-Alkhansari, and T. S. Huang, "A Fast Orthogonal Matching Pursuit Algorithm," IEEE International Conference of Acoustics, Speech, and Signal Processing (ICASSP), pp.1389-1392, 1998.

[4] S. Oh, and B. Jeon, "Fast Matching Pursuit using Symmetry of Absolute Property in Gabor Function," Conference of IPIU, Vol. 12, pp. 365-369, Jan. 2000.

[5] R. Neff and A. Zakhor, "Matching Pursuit Video Coding at Very Low Bit Rates," IEEE Data Compression Conf., Snowbird, UT, pp. 411-420, Mar. 1995.