NOVEL FAST BLOCK MOTION ESTIMATION IN FEATURE SUBSPACE

Yiu-Hung Fok*, Oscar C. Au** and Ross D. Murch***

Department of Electrical and Electronic Engineering
The Hong Kong University of Science and Technology
Clear Water Bay, Hong Kong

Email: eeyhfok@ee.ust.hk*, ecau@ee.ust.hk** and eermurch@ee.ust.hk***

ABSTRACT

Motion estimation and compensation are widely used in video coding. This paper presents two fast block matching algorithms for motion estimation. These algorithms use the subspace features of a block to determine the block distance. With a search block size of $N \times N$, the proposed algorithms can achieve a computation reduction factor of $N/2$ while retaining close-to-optimal performance in the mean absolute difference (MAD) sense.

1. INTRODUCTION

Video compression is vital and necessary for the transmission and storage of video sequences because of the huge data rate of the raw video data. There are many video compression standards, such as CCITT H.261 and MPEG, which use block-based motion estimation and compensation to reduce the temporal redundancy in video sequences. Conventionally, motion estimation is done by performing block matching exhaustively in the search area which is optimal with respect to the mean absolute difference (MAD) but it requires extremely large amount of computation. Therefore, different fast block matching algorithms had been proposed [1-7]. A fast block matching algorithm in feature domain (FBMA) was proposed in [1], which achieves a computation reduction factor by a factor of $N/2$ when compared with the conventional exhaustive search (CES) algorithm. Here the basic search blocks are of size $N \times N$. Although FBMA has close-to-optimal performance, it requires a large amount of additional memory which is undesirable for practical implementation.

In [1-3], simulations have shown that Re-examination can improve significantly the matching result. The idea is to use the MAD in the pixel domain given in equation (1) to re-examine the R best matching locations which are found by using another distance measure. The one with the least MAD in the pixel domain is declared the best match.

$$MAD_p(i, j) = \frac{1}{N^2} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} |f_k(x+m, y+n) - f_{k-1}(i+m, j+n)| \quad (1)$$

In this paper, two novel fast block matching algorithms without the additional memory requirement, Plus Pattern Search (PPS) and Cross Pattern Search (CPS), are presented. These algorithms make use of the

subspace features of the block to calculate the block distance so that the computation required for motion estimation can be reduced. Incorporated with Re-examination, the performance is found to be better than FBMA. With a block of size $N \times N$, these algorithms can achieve a computation reduction factor of $N/2$, which is significant, while having close-to-optimal performance in the MAD sense.

2. ALGORITHMS

The first is called Plus Pattern Search (PPS) and it is described below:

- Step 1. For an $N \times N$ block at the (x, y) -th location in the current k -th frame of size $Q \times P$, we define a search area of size $(2W+1) \times (2W+1)$ in its previous frame. Here the maximum horizontal and vertical displacement of the block is $\pm W$ pixels.
- Step 2. At each location (i, j) in the search area, we calculate the cost function using the Plus pattern features of the block as shown in Figure 1(a). The cost function is defined as the MAD of the Plus pattern features:

$$MAD_f = \frac{1}{K} \left\{ \sum_{n=0}^{N-1} |f_k(x+n, y+H) - f_{k-1}(i+n, j+H)| \right. \\ + \sum_{n=0}^{H-1} |f_k(x+H, y+n) - f_{k-1}(i+H, j+n)| \quad (2) \\ \left. + \sum_{n=H+1}^{N-1} |f_k(x+H, y+n) - f_{k-1}(i+H, j+n)| \right\}$$

where $f_k(x, y)$ is the pixel value at the (x, y) -th location of the k -th frame, $H = N/2 - 1$ and $K = 2N - 1$.

- Step 3. (Re-examination) We re-examine R best candidates found in Step 2 by calculating the MAD in the pixel domain defined in equation (1) and find the best match.

Figure 1: (a) Plus and (b) Cross pattern features.

The other new fast block matching algorithm is called Cross Pattern Search (CPS). This is similar to PPS. Both Step 1 and 3 are the same as PPS. The difference is in Step 2 and it is described below:

Step 2: At each location (i, j) in the search area, we calculate the cost function using the Cross pattern features of the block as shown in Figure 1(b). The cost function is defined as the MAD of the Cross pattern features:

$$MAD_f = \frac{1}{2N} \sum_{n=0}^H \{ |f_k(x+n, y+n) - f_{k-1}(i+n, j+n)| + |f_k(x+H-n, y+n) - f_{k-1}(i+H-n, j+n)| \} \quad (3)$$

where $H = N - 1$.

3. COMPUTATIONAL COMPLEXITY

For an $Q \times P$ frame, with a block size of $N \times N$ and maximum displacement of $\pm W$ pixels in both the horizontal and the vertical directions, the number of additive operations needed for motion estimation of one frame using CES is:

$$N_{CES} = \frac{PQ}{N^2} (2W+1)^2 (2N^2 - 1) \quad (4)$$

Let R be the number of locations used in re-examination in PPS and CPS. The additive operations needed for motion estimation of one frame are:

$$N_{PPS} = \frac{PQ}{N^2} [(2W+1)^2 (4N-3) + R(2N^2 - 1)] \quad (5)$$

$$N_{CPS} = \frac{PQ}{N^2} [(2W+1)^2 (4N-1) + R(2N^2 - 1)] \quad (6)$$

where the second term corresponds to the computation for Re-examination. For $W \geq N \gg R > 1$, the second term is insignificant compared with the first term. Consequently, the computation reduction factor of PPS and CPS is approximately:

$$\frac{N_{CES}}{N_{PPS}} = \frac{N_{CES}}{N_{CPS}} = \frac{2N^2 - 1}{4N - 3} \approx \frac{N}{2} \quad (7)$$

4. SIMULATION RESULTS

Our algorithms were simulated using the luminance component of 200 frames of the "Football" and the "Tennis" sequences. Each frame contained 352×240 pixels quantized uniformly to 8 bits. The size of the block was 16×16 . The maximum displacement in the search space was ± 16 pixels in both the horizontal and the vertical directions. For re-examination in each algorithm, we used $R=5$ best candidates in the feature domains.

Table 1 shows the percentage of MAD of different algorithms on the average larger than that of the CES and the percentage of Peak-Signal-To-Noise Ratio (PSNR) of different algorithms on the average smaller

than that of the CES. The PSNR of the estimated frame is defined as:

$$PSNR = -10 \log_{10} \frac{\sum_{y=0}^{P-1} \sum_{x=0}^{Q-1} [f(x, y) - \tilde{f}(x, y)]^2}{Q \times P \times 255^2} \quad (8)$$

where $f(x, y)$ and $\tilde{f}(x, y)$ are the original frame and the estimated frame of size $Q \times P$ respectively.

Algorithm	MAD larger than CES		PSNR smaller than CES	
	Football	Tennis	Football	Tennis
FBMA	2.85%	2.13%	0.70%	0.51%
PPS	4.66%	3.19%	1.93%	1.57%
CPS	1.87%	1.42%	0.66%	0.48%

Table 1: MAD larger than and PSNR smaller than CES.

Figure 2 shows the MAD between 30 estimated frames and 30 original frames of the "Football" sequence started from the 90th frame. Moreover, Figure 3 shows the MAD between 30 estimated frames and 30 original frames of the "Tennis" sequence started from the 50th frame. Figure 4 and 5 show the PSNR for the "Football" and "Tennis" sequences respectively. They show the general profiles of different algorithms.

As expected, the CES is optimal in the MAD sense. Simulation showed that, with a block of size $N \times N$, CPS performs better than both FBMA and PPS and it can achieve close-to-optimal MAD performance with a computation reduction factor of $N/2$ approximately.

Notice that, in 14 out of 200 frames of the "Football" sequence, the PSNR of CPS can even be larger than or equal to those of the CES. Moreover, in 32 out of 200 frames of the "Tennis" sequence, the PSNR of CPS are larger than or equal to those of the CES.

Figure 6 shows the estimated 82th frames of CES, FBMA, PPS and CPS of the "Football" sequence. In addition, Figure 7 shows the estimated 82th frames of CES, FBMA, PPS and CPS of the "Tennis" sequence. These frames are of lots of motions. In terms of subjective image quality, CPS performs better than PPS and FBMA. Also, the subjective image quality of CPS is very close to that of CES.

5. CONCLUSION

In this paper, we present two fast block matching algorithms using subspace features of the block for motion estimation, PPS and CPS. With block size of $N \times N$, these algorithms can achieve a computation reduction factor of $N/2$ which is significant. Simulations show that CPS can achieve close-to-optimal MAD performance with good subjective image quality. With

the regularity of the matching pattern and the close-to-optimal performance, CPS is a good substitution for conventional exhaustive search for motion estimation.

6. REFERENCES

- [1] Y. H. Fok and O. Au, "A Fast Block Matching Algorithm in Feature Domain," *Proc. of IEEE Workshop on Visual Signal Processing and Communications '93*, Melbourne, pp. 199-202, September 21-23, 1993.
- [2] Y. H. Fok and O. C. Au, "Improved fast block matching algorithm in feature domain," *Proc. of SPIE: Visual Communications and Image Processing '94*, Chicago, vol. 3, pp. 1826-1837, September 25-29, 1994.
- [3] Y. H. Fok and O. C. Au, "An Improved Fast Feature-based Block Motion Estimation," *Proc. of IEEE Int. Conf. Image Processing 1994*, Texas, vol. 3, pp. 741-745, Nov. 13-16, 1994.
- [4] B. Liu and A. Zaccarin, "New Fast Algorithms for the Estimation of Block Motion Vectors," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 3, No. 2, April 1993, pp. 148-157.
- [5] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Commun.*, vol. COM-29, December 1981, pp. 1799-1808.
- [6] S. Kappagantula and K. R. Rao, "Motion Compensated Interframe Image Prediction," *IEEE Trans. Commun.*, vol. COM-33, no. 9, September 1985, pp. 1011-1015.
- [7] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion compensated interframe coding for video conferencing," *Proc. Nat. Telecommun. Conf.*, New Orleans, L. A., Nov. 29-Dec. 3, 1981, pp. G5.3.1-5.3.5.

Figure 2: MAD for Football sequence.

Figure 3: MAD for Tennis sequence.

Figure 4: PSNR for Football sequence.

Figure 5: PSNR for Tennis sequence.

(a)

(b)

(c)

(d)

Figure 6: The Estimated 82th frame for Football sequence using (a)CES, (b)FBMA, (c)PPS and (d)CPS.

(a)

(b)

(c)

(d)

Figure 7: The Estimated 82th frame for Tennis sequence using (a)CES, (b)FBMA, (c)PPS and (d)CPS.