Fast Sub-pixel Motion Estimation by Means of One-Bit Transform

Orhan Akbulut, Oğuzhan Urhan, and Sarp Ertürk

Kocaeli University Laboratory of Image and Signal processing (KULIS),
Electronics and Telecom. Eng. Dept., University of Kocaeli, 41040, Turkey
orhanakbulut@gmail.com, {urhano, sertur}@kou.edu.tr

Abstract. In this study sub-pixel motion estimation using one-bit transform is presented. The utilization of low bit depth representations for full-pixel motion estimation exist in the literature [1,7]. However sub-pixel motion estimation employing these kinds of transforms has not been tried until now. In this paper, it is shown that one-bit transform can also be utilized for sub-pixel motion estimation, significantly improving motion estimation accuracy. Hence a fast sub-pixel motion estimation approach is facilitated using one-bit transform.

1 Introduction

Video compression enables effective utilization of both memory and bandwidth. Motion estimation and compensation are employed for exploiting temporal redundancy between consecutive image frames to obtain better compression performance. Efficient motion estimation and compensation enable effective compression of video data. However motion estimation comprises most of the coding time and computational complexity.

Block based motion estimation methods are generally used in video coding standards. In block based motion estimation, image frames are initially divided into blocks. Then, each block of the current frame is searched for in the previous frame within a pre-defined search window. The best matched block is regarded to be the motion compensated prediction and is subtracted from the current block to form the residual, and both the difference (residual) and motion vectors are encoded [2].

The full search strategy uses exhaustive search of all possible candidate locations within the search window, hence the best match is ensured. However, the computational load is extremely high. Many fast search methods have been proposed to reduce the computational burden of the full search approach, at the cost of lower accuracy. These methods commonly decrease the number of search points to reduce computational load. Some of the fast search approaches are three step search (3SS) [3], 2D logarithmic search (2DLOG) [4] and new three step search (N3SS) [5].

Several matching criteria have been proposed for block based motion estimation. Minimum absolute difference (MAD) is one of the error criteria commonly used for the matching process. Another matching criterion is the mean square error (MSE) which gives better Peak Signal to Noise Ratio (PSNR) according to other matching

A. Levi et al. (Eds.): ISCIS 2006, LNCS 4263, pp. 503 – 510, 2006.

criteria, but on the other hand, it is more difficult to implement the MSE criterion in hardware. Alternative matching error criteria for easier hardware calculation have been proposed in the literature.

Reduced bit resolution motion estimation methods have been proposed to obtain low-complexity matching criteria in [1,2,7]. In these cases, the matching criterion can be calculated faster using these low bit representation image frames using Boolean only operations. The exclusive-or (XOR) function is very efficient and can be easily implemented in hardware to perform matching criteria computation.

The one-bit (1BT) transform proposed in [1] is one of the first methods which uses low bit resolution representations. The main purpose of the 1BT transform is to represent image frames as binary images reducing the bit resolution. In this method, the 8-bit gray scale image frames are initially filtered using a multi band-pass filter kernel as given in (1).

$$K_{i,j} = \begin{cases} 1/25, & i,j \in [0,4,8,12,16] \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

After the filtering operation binary image frames ($G_{i,j}$) are obtained using (2) where $F_{i,j}$ denotes the original frame and $F'_{i,j}$ denotes the filtered frame obtained by filtering the original frame with the kernel K .

$$G_{i,j} = \begin{cases} 1, & F_{i,j} \geq F'_{i,j} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

The one-bit transform based motion estimation approach uses XOR (exclusive or) operation to compute block matching criteria. The calculation of the matching criteria for a given block is expressed as

$$\left\| G(t), G(t-1) \right\| = \sum_{i=1}^{N} \sum_{j=1}^{N} G_{i,j}(t) \otimes G_{i,j}(t-1) \tag{3}$$

where $\otimes$ and N show the XOR process and block size respectively. Hence, the 1BT transform uses the XOR process to perform block comparison. Therefore, a significant performance increment can be achieved in terms of speed in the case of hardware implementation of 1BT motion estimation approach. It is noted in [1] that the 1BT allows for a roughly 15:1 speed improvement with respect to the traditional architecture.

In this work 1BT based motion estimation method is modified for sub-pixel accuracy to improve the motion estimation performance. Firstly image frames are up-sampled by a factor of 2 and 4 for 0.5 and 0.25 pixel accuracy respectively. Appropriate kernels are utilized according to the image frame up-sampling factor to obtain 1-bit binary images. Then standard full search is employed using EXOR based matching criteria to obtain the motion vectors. Experimental results show that the proposed sub-pixel motion estimation approach via one-bit transform significantly outperforms the standard full-pixel 1BT based motion estimation approach, and can be used to facilitate fast sub-pixel motion estimation accuracy.

2 Sub-pixel Accuracy 1BT Matching

In block motion estimation, improved block matching results can be achieved if the motion vector is calculated with sub-pixel accuracy. MPEG2 and H.263 video coding standards support half (0.5) pel accuracy motion vectors. On the other hand, MPEG4 and H.264 standards support one-quarter (0.25) and one-eighth (0.125) pel accuracy motion vectors. Obviously better motion compensation accuracy can be obtained with higher resolution motion vectors. In other words, sub-pixel motion estimation can provide better compression than integer pixel motion estimation, at the expense of increased complexity.

In this work, the 6-tap FIR filter approach with tap values (1,-5, 20, 20,-5, 1) is used to up-sample image frames by 2 and 4 times, and the interpolation process is performed using this filter.

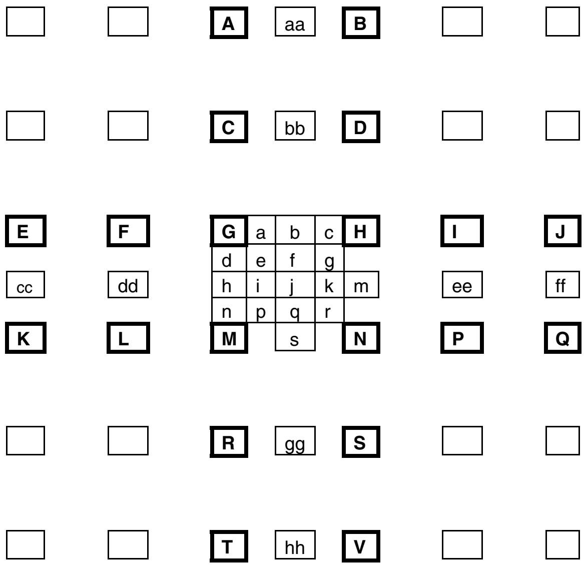

Fig. 1. Telenor up-sampling filter

Actually, several different interpolation filters exist in the literature such as the Bilinear filter, Wiener filter and Telenor filter [8]. The tap values of the filter and the filter length changes according to the desired filter structure. In this study the Telenor filter is used. The interpolation process with quarter-pel resolution consists of two stages. The first stage is interpolation to half-pel resolution. In other words, the image frame is up-sampled 2 times. In the second stage, the up-sampled image frame is again up-sampled by a factor of 2. As a result of this process, an interpolation with quarter-pel resolution is obtained. The Telenor filter is used to calculate new pixel values as shown in Figure1. Each pixel value is calculated as follows,

- Half-pel resolution pixel samples like 'b' are calculated by applying the 6-tap filter to the nearest integer position samples in the horizontal direction. Half-pel resolution pixel samples like 'h' are calculated by applying the 6-tap filter to the nearest integer position samples in the vertical direction.

$$b = (\text{E-5F+20G+20H-5I+J}) \\ \text{b}=((b+16)/32) \tag{4}$$

$$h = (\text{A-5C+20G+20M-5R+T}) \\ \text{h}=((h+16)/32) \tag{5}$$

- Half-pel resolution pixel samples like 'j' are calculated by applying the 6-tap filter to the nearest half-pel resolution samples in the horizontal direction or vertical direction.

$$j = cc - 5dd + 20h + 20m - 5ee + ff \\ j = aa - 5bb + 20b + 20s - 5gg + hh \tag{6}$$

$$\text{j}=((j+512)/1024) \tag{7}$$

- Quarter-pel resolution pixel samples like 'a,c,d,n,f,i,q,k' are calculated by averaging nearest integer position and nearest half-pel resolution pixel samples.

$$\text{a}=((\text{G+b})/2),\ \text{c}=((\text{H+b})/2) \\ \text{d}=((\text{G+h})/2),\ \text{n}=((\text{M+h})/2) \\ \text{k}=((\text{j+m})/2),\ \text{q}=((\text{j+s})/2) \\ \text{f}=((\text{b+j})/2),\ \ \text{i}=((\text{h+j})/2) \tag{8}$$

- Quarter-pel resolution pixel samples like 'e,g,p,r' are calculated by averaging two nearest half-pel resolution pixel samples in the diagonal direction.

$$\text{e}=((\text{b+h})/2),\ \text{g}=((\text{b+m})/2) \\ \text{p}=((\text{h+s})/2),\ \text{r}=((\text{m+s})/2) \tag{9}$$

For the 1BT based sub-pixel motion estimation process, first of all, the interpolation process is applied and frames are converted into binary bit plane representations similar to [1]. However different kernel sizes are used for half-pel and quarter-pel interpolated versions. Then, the block in the reference frame, is searched for in the 1BT of the previous frame using XOR matching.

An important point is that, if we use half-pixel resolution, the size of the maximum vector displacement that is used to determine the search window size, is increased by a factor of 2 (2:1 resolution). Similarly, if we use quarter-pixel resolution, the size of the maximum vector displacement is increased by a factor of 4 (4:1 resolution). Another important point is the size of the multi band-pass filter. The standard kernel filter given in (1) should not be applied on the interpolated image frames, and it is required to modify the filter kernel appropriately. Hence, new kernel filters (modified kernel matrices) are used in this application for the 1BTs of the up-sampled image frames. Mathematical expressions of the kernel filters are given in equations (10) and (11) respectively for 0.5 pixel and 0.25 pixel accuracy. The frequency response of the modified kernel matrix for half-pel accuracy motion estimation is shown in Figure 2.

$$K_{i,j} = \begin{cases} 1/25, & i,j \in [0,8,16,24,32] \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

$$K_{i,j} = \begin{cases} 1/25, & i,j \in [0,16,32,48,64] \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

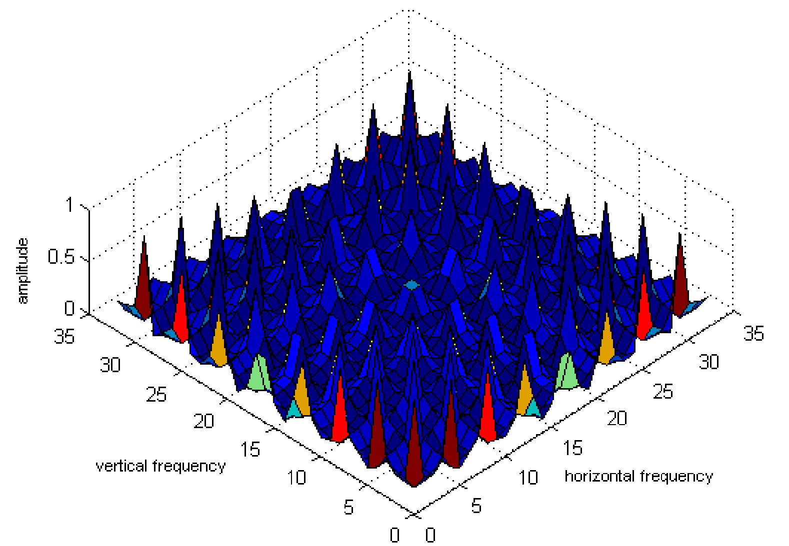

Fig. 2. Frequency response of modified filter kernel for half-pel accuracy motion estimation

3 Experimental Results

Several experiments are performed to evaluate the performance of the proposed sub-pixel motion estimation approach for various video sequences. Performance evaluation is made in terms of peak signal to noise ratio (PSNR) that is computed between the original frame and the reconstructed frame obtained from the previous frame using motion vectors computed by the proposed sub-pixel accuracy 1BT motion estimation scheme. PSNR values of the proposed approach with respect to frame numbers are given in Fig 3 for the first 100 frames of the "Foreman" sequence. It is clearly seen from this figure that the proposed sub-pixel based 1BT approaches generally outperform the standard full-pixel 1BT method proposed in [1], as expected.

Performance of the various motion estimation approaches for different sequences for block sizes of 16×16 and 8×8 are given in Table 1 and 2 respectively. MAD block matching criteria approach using sub-pel accuracy with full search (FS) is also performed and results are provided in these tables. As clearly shown from these tables sub-pixel accuracy motion estimation approach gives better performance than full-pixel MAD and full-pixel 1BT. Moreover, quarter-pixel accuracy 1BT approach provides up to 1.5 dB than the full-pixel 1BT for some test sequences. Quarter-pixel accuracy 1BT generally shows even better performance than the full-pixel 2BT method proposed in [7].

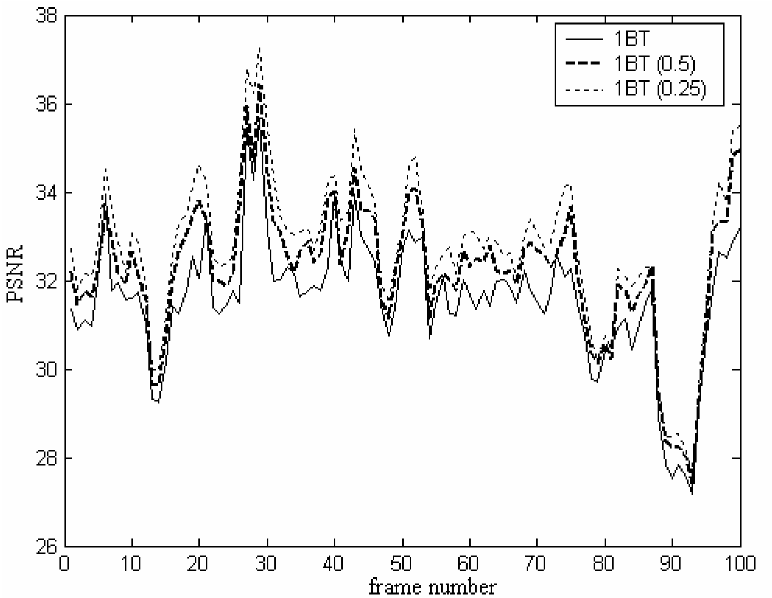

Fig. 3. PSNR values of the motion estimation methods for first 100 frames of the "Foreman" sequence at the block size of 8×8 pixels (1BT= One-bit [1], 1BT (0.5) = half-pel accuracy one-bit), 1BT (0.25) = Quarter-pel accuracy one-bit)

Table 1. Average PSNR values (dB) of several sequences for various motion estimation methods with block size of 16×16 pixels

	VIDEO SEQUENCES					
METHOD	Football (352x240 (125 frames)	Flower garden (352x240) (115 frames)	Mobile (352x240) (140 frames)	Tennis (352x240 (112 frames)	Coastguard (352x288) (299 frames)	Foreman (352x288) (299 frames)
MAD	22.88	23.79	22.99	29.87	30.48	32.11
MAD (0.5 pel)	23.50	24.86	24.87	30.60	31.21	33.56
MAD (0.25 pel)	23.82	25.52	25.80	30.98	31.82	34.19
2BT	22.08	23.43	22.72	28.89	29.93	30.71
1BT	21.83	23.32	22.71	28.77	29.84	30.44
1BT (0.5 pel)	22.31	24.15	24.49	29.27	30.36	31.13
1BT (0.25 pel)	22.53	24.61	25.27	29.49	30.79	31.54

Table 2. Average PSNR values (dB) of several sequences for various motion estimation methods with block size of 8×8 pixels

	VIDEO SEQUENCES					
METHOD	Football (352x240) (125 frames)	Flowergar-den (352x240) (115 frames)	Mobile (352x240) (140 frames)	Tennis (352x240) (112 frames)	Coastguard (352x288) (299 frames)	Foreman (352x288) (299 frames)
MAD	24.73	25.22	23.88	31.25	31.59	32.90
MAD (0.5 pel)	25.78	26.88	26.30	32.23	32.61	34.72
MAD (0.25 pel)	26.24	27.85	27.57	32.68	33.42	35.53
2BT	23.36	24.55	22.99	29.91	30.50	30.64
1BT	22.70	24.11	22.73	29.22	29.20	29.84
1BT (0.5 pel)	23.38	25.09	24.73	29.71	29.64	30.87
1BT (0.25 pel)	23.68	25.57	25.42	29.80	30.19	31.41

Comparing the performance of 1BT based sub-pixel motion estimation with conventional MAD sub-pixel motion estimation, it is seen that there is a performance drop of about 1 dB for a block size of 16×16 pixels, and a drop of about 2.5 dB for a block size of 8×8 pixels. As the interpolation scheme of both approaches is the same, and the gain in 1BT is low-complexity matching of single bit-planes, the speed improvement can be regarded as 15:1 in accordance with [1].

4 Conclusion

Low bit resolution representations have been presented in the literature for motion estimation [1,7]. However sub-pixel motion estimation employing these kinds of transforms has not been tried until now. In this work, sub-pixel motion estimation via one-bit transform using interpolation of image frames followed by 1BT is proposed. Experimental results show that the proposed approach outperforms standard full-pixel 1BT method in terms of PSNR. Furthermore, better PSNR results are obtained than full-pixel 2BT for quarter-pel accuracy of 1BT. Fast search approaches can be used with 1BT to further decrease the computational load at sub-pixel level.

References

[1] B. Natarajan, V. Bhaskaran, and K. Konstantinides, "Low-Complexity Block-Based Motion Estimation via One-Bit Transforms," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 7, No. 4, pp. 702-706, Aug. 1997.

[2] P. H. W. Wong and O. C. Au, “Modified One-Bit Transform for Motion Estimation,” IEEE Trans. Circuits Syst. Video Technol., Vol. 9, No. 7, pp. 1020-1024, Oct. 1999.

[3] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, “Motion-Compensated Interframe Coding for Video Conferencing,” Proc. of Nat. Telecommun.Conf., pp. G5.3.1-5.3.5, Nov./Dec. 1981.
[4] J. R. Jain and A. K. Jain, “Displacement Measurement and Its Application in Interframe Image Coding,” IEEE Trans. Communication., Vol. COM-29, pp. 1799-1808, Dec. 1981.
[5] R. Li, B. Zeng, and M. L. Liou, "A New Three-Step Search Algorithm for Block Motion Estimation," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 4, No. 4, pp. 438-442, Aug. 1994.
[6] A. M. Tekalp, Digital Video Processing. Englewood Cliffs, NJ: Prentice-Hall, 1995, pp. 432-455.
[7] A. Ertürk and S. Ertürk, “Two Bit Transform for Binary Block Motion Estimation ” IEEE Trans. On Circuits and Systems for Video Technology, Vol. 15, No. 7, pp. 938-946, July 2005.
[8] G. Bjontegaard, “Motion Compensation with 1/4 pixel accuracy”, ITU-T SG16/Q15, Geneva, Feb. 2000.