Constrained One-Bit Transform for Low-Complexity Block Motion Estimation

Oğuzhan Urhan, *Member, IEEE,* Sarp Ertürk, *Member, IEEE*

***Abstract*— One-bit transform (1BT) and two-bit transform (2BT) based block motion estimation schemes have been proposed in the literature to reduce the computational complexity of the motion estimation process by enabling simple Boolean EX-OR matching of lower bit-depth representations of image frames. Recently a multiplication-free one-bit transform (MF-1BT) has been proposed to facilitate 1BT to be carried out with integer arithmetic using addition and shifts only. Thresholding schemes are typically used in order to construct the lower bit-dept representations utilized in 1BT and 2BT. In our experience we have observed that one problem with such schemes is that pixel values that lie on directly opposite sides of the threshold are categorized into separate classes and are therefore counted as a non-match in the search process even if they are close in value. A constrained 1BT (C-1BT) that restricts pixels with values adjacent to the transform threshold during 1BT matching, counting them as a match regardless of their 1BT value, is proposed in this paper. It is shown that the proposed C-1BT approach improves the motion estimation accuracy of 1BT based motion estimation and even outperforms 2BT based motion estimation at macroblock level.**

***Index Terms*—- Motion estimation, One-bit transform, Two-bit transform, Image/video coding and transmission**

I. Introduction

Motion estimation (ME) is usually remarked as the computationally most intensive part of the video compression system, performing up to 50 % of the computations executed by the entire video coding system [1]. The high computational load might not be a problem for applications that can use offline processing and can encode the video in advance, such as it is the case for broadcast applications for instance. However, due to the increase in portable and wireless equipment that have video recording or transmission capabilities, it has become important to reduce the computational complexity of video compression schemes, and in particular the motion estimation process, for some applications. Techniques proposed to reduce the high computational load of the full search minimum absolute difference, FS-MAD, block matching algorithm can mainly be divided into three categories [2]: fast search techniques that select a subset of the possible search candidate locations; techniques based on various forms of pixel pattern or motion field decimation that employ a certain sub-sampling of the pixel pattern or motion field; and techniques that exploit different matching criteria instead of the classical MAD.

One-bit transform (1BT) and two-bit transform (2BT) techniques have been proposed to facilitate Boolean Exclusive-OR (EX-OR) matching of low bit-depth representations of image frames so as to exploit different matching criteria to achieve reduction in computational complexity. Bit-plane matching as a preprocessing step to exhaustive search has been proposed in [3] to eliminate unlikely locations, with the block mean being used as threshold to accomplish a one-bit transform. The resulting bit-plane of an image frame is obtained in the form of

$$B(i,j)=\begin{cases}1\,, & \text{if}\, I(i,j)\geq t_{bm}\\ 0\,, & \text{otherwise}\end{cases} \tag{1}$$

where t_{bm} represents the threshold value that is set equal to the block mean, and (i,j) is used as pixel index. Hierarchical feature matching-motion estimation (HFM-ME) that employs sign truncated feature (STF) matching has been proposed in [4]. Binary block matching on the binary edge maps of image frames has been proposed in [5], however it has been noted that the technique is inappropriate for blocks with inadequate edge information. In [6] motion estimation using the one-bit transform (1BT), where image frames are transformed into one-bit/pixel representations by comparing the original image frame against a multi band-pass filtered version is proposed. A 17×17 multi band-pass filter kernel in the form of

$$K(i,j)=\begin{cases}1/25\,, & \text{if}\, i,j\in[1,4,8,12,16]\\ 0\,, & \text{otherwise}\end{cases} \tag{2}$$

is used to filter the image frame, and the one-bit transform bit-plane of an image is constructed as

$$B(i,j)=\begin{cases}1\,, & \text{if}\, I\left(i,j\right)\geq I_F\left(i,j\right)\\ 0\,, & \text{otherwise}\end{cases} \tag{3}$$

O. Urhan and S. Ertürk are with Kocaeli University Laboratory of Image and Signal processing (KULIS), Electronics and Telecom Eng. Dept., University of Kocaeli, 41040, TURKEY.

where $I_F(i,j)$ represents the filtered version of the image frame $I(i,j)$, obtained by filtering I with the convolution kernel K. Because the filtered version is compared against the original image, to construct the representative bit-plane, the filtered image is mainly used as a sort of pixel-wise threshold. In other words, the 1BT is constructed by comparing the original image against the threshold pattern obtained by applying the image to a multi band-pass filter.

For motion estimation, the single bit-plane representation (i.e. one-bit transform) of each image frame is constructed as shown in (3), and the motion vector of a block is decided based on the number of non-matching points (NNMP) measure:

$$\begin{aligned} &NNMP(m,n)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left\{B^{t}(i,j)\oplus B^{t-1}(i+m,j+n)\right\} \\ &-s\leq m,n\leq s-1 \end{aligned} \tag{4}$$

where (m,n) shows the candidate displacement, s determines the search range, and $\oplus$ shows the exclusive-OR (EX-OR) operation. The candidate displacement that gives the lowest NNMP is designated to be the block motion vector. If two candidate displacements result in the same NNMP value, the one with the lowest distance to the block position (i.e. the smallest motion vector) is selected.

The utilization of this 1BT based motion estimation approach for low-complexity digital image stabilization with possible applications in camcorders or wireless video communications equipment has been reported in [7]. The addition of conditional local searches has been proposed in the modified one-bit transform to improve the predicted image at the expense of increased computational complexity in [8], however in this case the binary only matching characteristics is destroyed as additional local MAD matches have to be executed. A two-bit transform (2BT) that makes use of local mean and variances to construct regional thresholds so as to obtain a two-bit representation has been proposed in [9] and a DCT-based adaptive thresholding approach for binary motion estimation has been proposed in [10] to improve motion estimation accuracy at the cost of higher computational complexity. Recently a multiplication-free one-bit transform (MF-1BT) approach has been proposed in [11] to further reduce the computational load of the 1BT process itself.

In our experience we have observed that one factor that reduces the accuracy of 1BT schemes is the fact that pixel values on directly opposite sides of the threshold are categorized into separate classes and counted as a non-match in the search process even if original pixel values are actually close. It is proposed in this paper to exclude pixels with values close to the transformation threshold from 1BT matching, i.e. force them to be counted as a match regardless of their 1BT values, to overcome the aforementioned consequence. The proposed approach is referred to as constrained one-bit transform (C-1BT), and makes use of the MF-1BT to achieve overall low transform complexity.

II. CONSTRAINED ONE-BIT TRANSFORM

Floating point multiplication operations in the 1BT proposed in [6] cannot be avoided, because the normalization coefficient of the kernel shown in (2) is not a power of two. However, floating point multiplications are slow in both, hardware and software implementations [12]. A novel diamond shaped 1BT filtering kernel that also performs multi band-pass filtering has been proposed in [11]. The number of ones in this kernel is adjusted to be a power of two, and because the normalization coefficient is now a power of 2, it becomes possible to carry out this filtering by simply adding the corresponding 16 pixels and shifting the final result. Hence the complete filtering operation can be carried out using integer arithmetic and the computational load is 16 addition and 1 shift operations per pixel.

In the 1BT proposed in [11] the new diamond shaped convolution kernel is used to filter image frames to construct the 1BT using (3). As a result of the 1BT, image frames are converted into a single bit-plane representation. While this representation is generally successful in discriminating pixels with different values, one aspect that reduces the motion estimation accuracy is the fact that pixel values on directly opposite sides of the threshold are categorized into separate classes and counted as a non-match in the matching process, even if original pixel values are actually close. This case is illustrated in Figure 1. Here, T denotes the transform threshold which is basically equivalent to $I_F(i, j)$ in 1BT. In this case, pixels a and b are above the threshold so that their corresponding 1BTs can be denotes as $B(a) = 1$ and $B(b) = 1$. On the other hand pixels c and d are below the threshold so that $B(c) = 0$ and $B(d) = 0$. Hence if 1BT matching using the NNMP in (4) is used directly, a and b as well as c and d are counted as a match because $B(a) \oplus B(b) = 0$ and $B(c) \oplus B(d) = 0$. All other combinations are counted as a non-match because $B(a) \oplus B(c) = 1$, $B(a) \oplus B(d) = 1$, $B(b) \oplus B(d) = 1$ and also $B(b) \oplus B(c) = 1$. The problem here is that pixel values b and c are counted as a non-match because they are on opposite sites of the threshold (as a result of which they get different 1BT values) although they are close in value and it would be more correct to count them as a match.

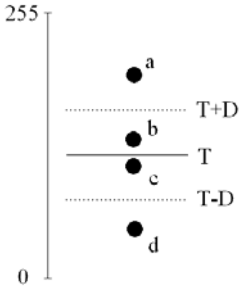


Figure 1. Sample pixel values (a,b,c,d) shown in the range of [0,255] to illustrate the influence of the distance to the 1BT transform threshold.

It is proposed in this paper to constrain pixels that have values close to the transform threshold, so that these pixels are excluded from the 1BT matching process. In other words, pixels that have values close to the transform threshold will be counted as a match regardless of their 1BT value. For this purpose, it is proposed to construct a constrain mask (CM) in the form of

$$CM(i,j)=\begin{cases}1, & \text{if } \left|I(i,j)-I_F(i,j)\right|\geq D \\ 0, & \text{otherwise}\end{cases} \tag{5}$$

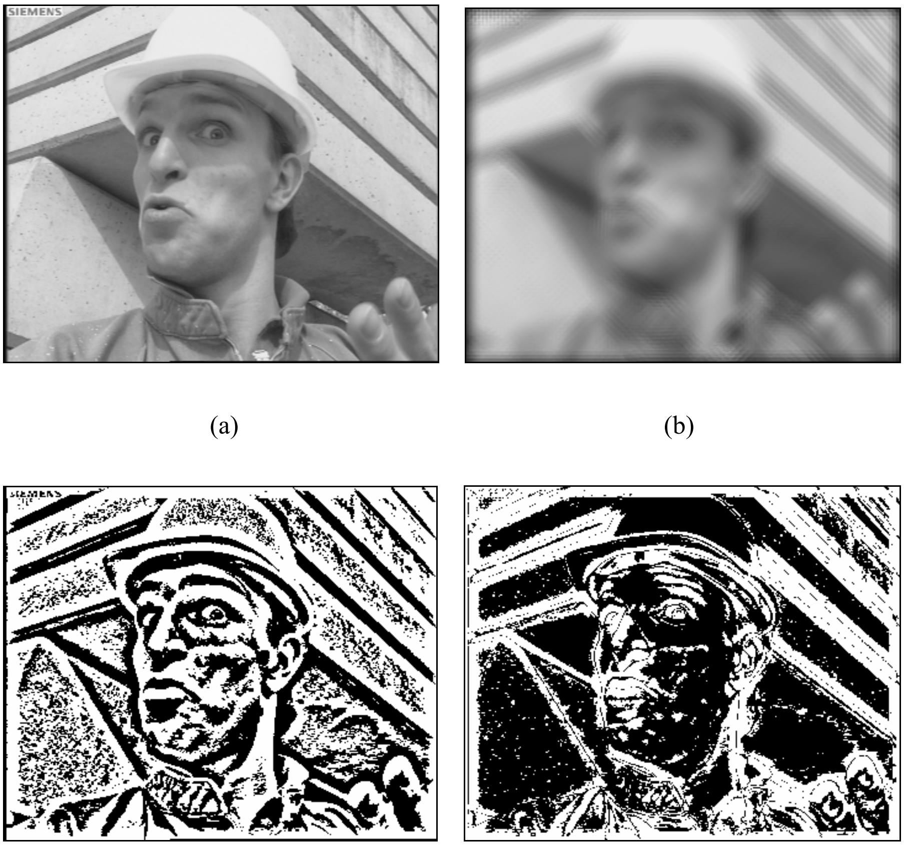
so that the CM value of the pixel will be 1 if the pixel value is at least a certain distance (*D*) away from the transform threshold (which is actually defined by I_F) to indicate that this pixel can reliably be included in the 1BT matching process. Otherwise the CM value of the pixel is set to zero to indicate that this pixel should be excluded from the 1BT matching process. Note that the 1BT result obtained using (3) shows already if $I(i,j)$ is greater than $I_F(i,j)$ or not. Hence it actually indicates whether $I(i,j)-I_F(i,j)$ will be positive or negative, and can be used to aid the absolute operation in the constrain mask.

The search process of the motion estimation is modified to take the constrain mask into account. The constrained number of non-matching points (CNNMP) measure is defined as shown in (6). Note that $||$ shows Boolean OR and $\odot$ shows Boolean AND operations.

$$\begin{aligned}&CNNMP(m,n)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left\{\left[CM^{t}(i,j)\,||\,CM^{t-1}(i+m,j+n)\right]\odot\left[B^{t}(i,j)\oplus B^{t-1}(i+m,j+n)\right]\right\}\\&-s\leq m,n\leq s-1\end{aligned} \tag{6}$$

For the pixel values shown in Figure 1 it is seen that pixels b and c are close to the transform threshold so that $CM(b)=0$ and $CM(c)=0$, while pixel values a and d are further away so that $CM(a)=1$ and $CM(d)=1$. It is seen from the definition of the $CNNMP$ measure in (6) that if at least one of the two pixels has constrain mask value of unity the 1BT matching result will decide if the two pixels are a match or not. On the other hand, if both pixels have CM values of zero, the 1BT matching result will be ineffective, and the total contribution of these pixels to the $CNNMP$ measure will be zero, in other words these pixels will be counted as a match. For the pixel values depicted in Figure 1 it is seen that pixels b and c will be counted as a match in the proposed $CNNMP$ measure, while the matching results of the other pixel values is not affected by the introduced CM and the 1BT matching result will be decisive.

Figure 2 shows a sample image frame, the filtered version, and the corresponding one-bit transform as well as the constrain mask. It is seen from the constrain mask that pixels with values close to the filtered version are indicated to be unreliable so as to exclude them during the 1BT matching process. This situation is observed to be encountered more frequently in smooth (i.e. low spatial frequency) parts of the image.



(a) (b)

(c) (d)

Figure 2. (a) A sample frame of the Foreman sequence, (b) Filtered version using the kernel shown in (5), (c) The corresponding one-bit transform obtained with MF-1BT, (d) The corresponding constrain mask.

The proposed C-1BT actually constructs two different bit-planes for each pixel, one for the 1BT and one for the CM. While it might seem that the proposed approach is more similar to the 2BT in this aspect, an important difference is that in 2BT both bit-planes are used for the search and matching process to indicate a reasonable match (the video frame is typically segmented into four separate classes in 2BT), and hence both bit-planes have the same purpose. On the other hand, in the proposed approach the CM is used to indicate if the corresponding 1BT should be included in the matching process or not. Although two bit-planes are constructed in total, these are not used to segment the frame into four classes as it is done in the 2BT approach.

Table I shows the number of operations needed per pixel (pp) to evaluate the computational complexity, and it is seen that the computational load of the matching process for the proposed approach given in (6), is equivalent to the computational load of the matching process of 2BT. An important advantage of the C-1BT is that it can be carried out using integer arithmetic only with addition, subtraction and shift operations without the need for multiplication, which enables lower computational complexity compared with the 2BT approach. It is stated in [6] that the 1BT proposed in [6] allows for a roughly 15:1 speed improvement with respect to the traditional architecture (i.e. MAD). The multiplication free and integer arithmetic nature of MF-1BT and C-1BT make these approaches particularly suitable for hardware implementations.

	Transform						Matching	Memory
	Addition (pp)	Multiplication (pp)	Shift (pp)	Subtraction (pp)	Comparison (pp)	Boolean Operation (pp)	Boolean Operation (pp)	Bits (pp)
1BT [6]	25	1	-	-	1	-	1	1
2BT [9]	2.8125	1.0625	-	0.03125	3	1	3	2
MF-1BT[11]	16	-	1	-	1	-	1	1
C-1BT	16	-	1	1	2	-	3	2

Table I. Number of operations needed for the low-complexity motion estimation schemes.

III. Experimental Results

In order to evaluate the performance of the proposed approach, motion estimation has been performed for various video sequences using exhaustive full search. For the matching criteria; MAD matching, 2BT as proposed in [9], 1BT as proposed in [6], MF-1BT as proposed in [11] and the proposed constrained 1BT (C-1BT) are evaluated. Note that other one-bit transforms proposed in the literature are not evaluated for comparison as the procedures of [6] and [11] are mainly the most accurate and simple one-bit transform approaches. In order to evaluate the motion estimation accuracy, initially, peak signal to noise (PSNR) values obtained between the original frames and frames reconstructed from the previous frame using motion vectors calculated from one of the methods, are used. The reason for using such an open loop scheme is to initially evaluate the accuracy of motion estimation by itself. The motion estimation accuracy is assessed on a macroblock basis for a block size of 16×16 pixels with a search range of 16 pixels, and the corresponding PSNR results for various test sequences are given in Table II. Note that the distance value D used in the CM of (6) for the proposed C-1BT is set to 10 in this case, and this value has been determined experimentally. It is seen the proposed C-1BT provides higher motion estimation accuracy compared to both, 1BT as well as MF-1BT. The results are mostly within 1dB of MAD, as sufficient detail can be captured within each block at this block size. While 2BT slightly outperforms 1BT and MF-1BT, it is seen that the proposed C-1BT performs similar to 2BT for some sequences and even outperforms 2BT for some.

Method	Video Sequences (Frame Size, Sequence Length)					
	Football (352×240) (125 frames)	Foreman (352×288) (300 frames)	Tennis (352×240) (150 frames)	Garden (352×240) (115 frames)	Mobile (352×240) (300 frames)	Coastguard (352×288) (300 frames)
MAD	22.88	32.09	29.45	23.79	23.94	30.48
2BT [9]	22.06	30.70	28.46	23.43	23.66	29.94
1BT [6]	21.83	30.32	28.11	23.31	23.61	29.83
MF-1BT [11]	21.81	30.38	28.18	23.26	23.63	29.88
C-1BT	22.10	30.86	28.71	23.38	23.69	29.98

Table II. Average PSNR (dB) of several sequences reconstructed by various motion estimation techniques, with full search and a block size of 16×16 pixels with a motion vector search range of 16 pixels.

In order to evaluate the sensitivity of C-1BT to the distance value D, Table III shows the PSNR results obtained in the open-loop scheme for various D values. It is seen from Table III that the optimal D value actually changes from sequence to sequence. For the Football sequence, which is regarded as a high-motion video [13], it is seen that a larger D value provides slightly higher PSNR, and thus improved motion estimation accuracy. On the contrary for the Foreman and Coastguard sequences, which are regarded as medium-motion videos [13], it is seen that PSNR values reduce for larger D values. The Mobile and Garden sequences show some irregular motion such as rotation [13], and it is seen that in this case there is nearly no change in PSNR results for changes in the D value. In overall $D = 10$ is found to provide acceptable results for all sequences.

	Video Sequences (Frame Size, Sequence Length)					
	Football (352×240) (125 frames)	Foreman (352×288) (300 frames)	Tennis (352×240) (150 frames)	Garden (352×240) (115 frames)	Mobile (352×240) (300 frames)	Coastguard (352×288) (300 frames)
C-1BT, $D=6$	21.98	31.01	28.48	23.35	23.67	29.98
C-1BT, $D=8$	22.04	30.96	28.61	23.36	23.68	29.98
C-1BT, $D=10$	22.10	30.86	28.71	23.38	23.69	29.98
C-1BT, $D=12$	22.14	30.70	28.73	23.40	23.70	29.92
C-1BT, $D=14$	22.19	30.48	28.72	23.41	23.71	29.81

Table III. Average PSNR (dB) of several sequences reconstructed by using motion vectors obtained by C-1BT using different distance values.

In order to evaluate the performance of the proposed ME approach in a video codec, the H.263+ reference software tmn3.2 is utilized. Tables IV and V provide the PSNR results for the Coastguard and Foreman sequences of QCIF size and with 300 frames encoded at 10 frames/sec at different bit-rates with various ME schemes. The Coastguard sequence has mainly high-spatial frequency content, and 1BT approaches already provide reasonable results due to this feature with only minor improvements gained by 2BT or C-1BT as seen in Table II. On the other hand, the Foreman sequence has some low-spatial frequency content, and in this case 2BT and C-1BT approaches can improve the 1BT performance more effectively as observed from Table II. Table IV shows that for the Coastguard sequence C-1BT provides slightly below 2BT, but superior than 1BT and MF-1BT, particularly at low bit-rates. As the bit-rate increases, the video coder can more effectively encode the prediction error, so that there is only a small difference between MAD and low complexity ME matching approaches at higher bit-rates. Table V shows that for the Foreman sequence C-1BT considerably outperforms 1BT and MF-1BT as well as 2BT, and the main reason is that frames of this sequence have some low-spatial frequency content for which the constrain feature proposed in this paper becomes particularly effective. Again, the PSNR gap between MAD and low complexity ME matching reduces with increasing bit-rate.

	Bit-rate (kbit/sec)										
	50	75	100	125	150	175	200	225	250	275	300
MAD	27.12	28.62	29.73	30.51	31.30	32.08	32.82	33.47	34.06	34.66	35.20
1BT [6]	26.04	27.81	29.09	30.03	30.91	31.74	32.51	33.18	33.80	34.39	34.94
MF-1BT [9]	26.11	27.82	29.16	30.13	31.00	31.82	32.55	33.26	33.88	34.46	35.00
2BT [11]	26.65	28.07	29.34	30.31	31.14	31.97	32.73	33.37	34.01	34.60	35.14
C-1BT	26.40	27.98	29.22	30.17	31.05	31.87	32.63	33.32	33.95	34.56	35.11

Table IV. Average Y-PSNR (dB) of the Coastguard sequence for H263+.

	Bit-rate (kbit/sec)										
	50	75	100	125	150	175	200	225	250	275	300
MAD	28.57	30.34	31.67	32.64	33.62	34.48	35.29	35.99	36.65	37.27	37.83
1BT [6]	27.15	29.19	30.68	31.89	32.89	33.84	34.68	35.41	36.08	36.72	37.31
MF-1BT [9]	27.45	29.31	30.86	32.05	33.08	34.04	34.88	35.62	36.31	36.94	37.52
2BT [11]	27.39	29.20	30.71	31.94	32.98	33.98	34.82	35.57	36.26	36.88	37.47
C-1BT	27.83	29.62	31.16	32.31	33.32	34.28	35.10	35.83	36.52	37.16	37.73

Table V. Average Y-PSNR (dB) of the Foreman sequence for H263+.

Experimental results demonstrate that the proposed C-1BT performs similar to, or even better than 2BT and certainly better than previous 1BT approaches. Although it might be possible to further improve the performance of C-1BT using an adaptive distance measure D in the computation of the constrain mask, the computational complexity of this procedure is of main concern, as the aim is to provide a low complexity ME approach. Note that it is also possible to extend 1BT and 2BT based ME approaches to sub-pixel level in a straightforward way to obtain improved accuracy at sub-pixel level as is shown in [14], by applying the transform after spatial interpolation of video frames.

IV. Conclusions

This paper proposed to constrain the pixels utilized in the matching of 1BT based low-complexity block motion estimation to enhance motion estimation accuracy. The proposed approach is referred to as constrained one-bit transform (C-1BT) based motion estimation. An important feature of the proposed C-1BT is that it does not require multiplication operations and can be implemented using integer arithmetic with addition, subtraction and shifts only. The proposed approach improves the performance of 1BT based motion estimation schemes using simple and low complexity operations. Experimental results show that the proposed approach gives improved motion estimation accuracy compared to 1BT [6], MF-1BT [11] as well as 2BT [9]. The proposed approach is particularly suitable for applications that require portable and mobile video encoding due to the low complexity and low power consumption needs in such cases.

References

[1] Z.-L. He, C.-Y. Tsui, K.-K. Chan, and M.L.Liou, “Low-power VLSI design for motion estimation using adaptive pixel truncation”, IEEE Trans. Circuits Syst. Video Technol., vol.10, pp. 669-678, Aug. 2000.

[2] M. Mattavelli, and G. Zoia, “Vector-tracing algorithm for motion estimation in large search windows”, IEEE Trans. Circuits Syst. Video Technol., vol.10, pp. 1426-1437, Dec. 2000.

[3] J. Feng, K.-T. Lo, H. Mehrpour, A. E. Karbowiak, “Adaptive block matching motion estimation algorithm using bit-plane matching”, in Proc. ICIP’95, pp. 496 - 499, 1995.

[4] X. Lee and Y. Zhang, “A Fast Hierarchical Motion-Compensation Scheme for Video Coding Using Block Feature Matching”, IEEE Trans. Circuit Syst. Video Technol., vol. 6, pp. 627 - 635, Dec. 1996.

[5] M. M. Mizuki, U. Y. Desai, I. Masaki, and A. Chandrakasan, “A binary block matching architecture with reduced power consumption and silicon area requirement”, in Proc. IEEE ICASSP-96, Atlanta, vol. 6, pp. 3248-3251,1996.

[6] B. Natarajan, V. Bhaskaran, and K. Konstantinides, “Low-complexity block-based motion estimation via one-bit transforms”, IEEE Trans. Circuit Syst. Video Technol., vol. 7, pp. 702 - 706, Aug. 1997.

[7] A.A. Yeni, and S. Ertürk, “Fast digital image stabilization using one bit transform based sub-image motion estimation,” IEEE Trans. Consumer Electronics, vol. 51, no. 3, pp. 917-921, Aug. 2005.

[8] P. H. W. Wong and O. C. Au, “Modified One-Bit Transform for Motion Estimation”, IEEE Trans. Circuit Syst. Video Technol., vol. 9, pp. 1020 - 1024, Oct. 1999.

[9] A. Ertürk, and S. Ertürk, “Two-Bit Transform for Binary Block Motion Estimation”, IEEE Trans. Circuit Syst. Video Technol., vol. 15, no. 7, pp. 938- 946, Jul. 2005.

[10] C.-B. Wu, C.-Y. Yao, B.-D. Liu, and J.-F. Yang, “DCT-Based Adaptive Thresholding Algorithm for Binary Motion Estimation”, IEEE Trans. Circuit Syst. Video Technol., vol. 15, no. 5, pp. 694- 703, May 2005.

[11] S. Ertürk, “Multiplication-free one-bit transform for low-complexity block-based motion estimation”, IEEE Signal Processing Letters, Feb. 2007, Accepted for Publication.

[12] J. Liang, and T. D. Tran, “Fast Multiplierless Approximations of the DCT With the Lifting Scheme” , IEEE Trans. Circuit Syst. Video Technol., vol. 49, no. 12, pp. 3032- 3044, Dec 2001.

[13] X. Yi, and N. Ling, “Rapid block-matching motion estimation using modified diamond search algorithm”, IEEE International Symposium on Circuits and Systems, ISCAS 2005, vol. 6, pp. 5489- 5492, May 2005.

[14] O. Akbulut, O. Urhan, and S. Ertürk, “Fast Sub-Pixel Motion Estimation by means of One-Bit Transform”, Lecture Notes in Computer Science (LNCS), Springer-Verlag, vol. 4263, pp. 503-510, Nov 2006.