Low Complexity Multiple Candidate Motion Estimation Using Gray Coded Bit-plane Matching

Changryoul Choi and Jechang Jeong
Dept. of Electronic and Communication Engineering Hanyang University
Seoul, Korea
e-mail: denebchoi@gmail.com & jjeong@hanyang.ac.kr

***Abstract*—In this paper, we propose low complexity motion estimation algorithms based on the Gray-coded bit-plane matching. By exploiting almost identical operations among similar but different matching error criteria, we can efficiently determine the respective candidate motion vectors. In addition, adopting multiple candidate motion estimation strategies into those candidate motion vectors and local searches around the best candidate motion vector dramatically enhance the motion estimation accuracy. Experiments were carried out for comparing the performances of the proposed algorithms with other bit-plane matching based motion estimation algorithms and full search block matching algorithm with the sum of absolute differences as well. Surprisingly, the peak signal to noise ratio difference between one of the proposed algorithms and the full search block matching algorithm is within 0.05dB on average.**

Keywords-motion estimation; bit-plane matching; video coding

I. INTRODUCTION

Due to the rapid growth of the multimedia service, the video compression has become essential for reducing the required bandwidth for transmission and storage in many applications. In video compression, the motion estimation (ME) and the motion compensation (MC) is the most crucial part since it can reduce the total video data efficiently by exploiting the temporal correlation between neighboring frames. The block matching algorithm (BMA) is adopted in many video compression standards because of its simplicity and effectiveness [1][2]. In BMA, a current frame is partitioned into small square (possibly rectangular) blocks and a motion vector is estimated within its search range in the reference frame by searching the most similar block according to some matching criterion such as the sum of absolute differences (SAD). Although the full search BMA (FSBMA) can find an optimal motion vector according to some matching criterion, its computational complexity is so huge that it is not adequate for real time applications. Therefore, many techniques including the fast searching and the fast matching algorithms have been proposed to reduce the high computational complexity of the FSBMA in the literature. Among them, there are some techniques that use different matching criteria instead of the classical SAD to make faster the computation of the matching criterion itself exploiting the bit-wise operations [3]-[12]. These algorithms are called bit-plane matching (BPM) based ME. The advantages of these techniques over the matching algorithms using the classical SAD are two-fold: fast computation of the matching criterion and reduced memory bandwidth in the interim of ME process. These techniques include one-bit transform (1BT) [3], multiplication-free 1BT [4], constrained 1BT (C1BT) [5], two-bit transform (2BT) [6], weighted 2BT [9], truncated Gray-coded BPM (TGCBPM) [7][12], weightless TGCBPM (WTGCBPM) [8], etc. Among the above algorithms that use bit-wise operations based matching criterion, TGCBPM and WTGCBPM show the best results in terms of the ME accuracy. In addition, the transforming the frame into bit-planes is relatively simple.

Although the BPM based ME succeeded in reducing the computational complexity, its ME accuracy is relatively poor compared with the SAD based ME, resulting in degraded reconstructed images. To remedy this situation, multiple candidate motion searches were proposed in [11]. In general, performing many motion searches using different matching criteria simultaneously helps improving the overall performance, but their computational complexities would increase heavily with the number of matching criteria used. However, they exploited correlation between two different matching criteria and did the dual motion searches with negligible computational complexity increase.

In this paper, we propose low complexity ME algorithms based on the Gray-coded BPM. By exploiting the similar operations between different matching error criteria, we can efficiently determine the respective candidate motion vectors. Adopting the multiple candidate motion search strategy into those candidate motion vectors and local searches around the best candidate motion vector dramatically enhance the motion estimation accuracy.

The rest of this paper is organized as follows. In Section 2, we review some previous works related to the proposed algorithm. In Section 3, we explain our proposed algorithm. Experimental results and analyses are presented at Section 4. Finally, Section 5 provides our conclusions.

II. PREVIOUS ALGORITHMS

In this section, we review some previous works on BPM based ME and multiple candidate motion search.

A. Bit-plane Matching Based Motion Estimation

In [12], Gray-coded BPM based ME was first proposed. And, TGCBPM and its variation WTGCBPM were proposed in [7][8]. Both TGCBPM and WTGCBPM use Gray-code mapping as transforming image frames into bit-planes, which is very simple compared to other bit-plane transformation algorithms using complex filtering operations (e.g., 1BT, 2BT, C1BT, etc.). Let the gray-level of the pixel at location (m, n) be represented as follows:

$$f(m,n) = a_{K-1}2^{K-1} + a_{K-2}2^{K-2} + \cdots + a_1 2^1 + a_0 2^0 \quad (1)$$

where a^k ($0 \le k \le K$-1) takes on either 0 or 1. Then the corresponding Gray-code representation is given by

$$\begin{aligned} g_{K-1} &= a_{K-1} \\ g_k &= a_k \oplus a_{k+1}, \quad 0 \le k \le K-2 \end{aligned} \quad (2)$$

where $\oplus$ denotes the Boolean XOR operation and a_k is the k-th bit representation. This Gray code representation has the unique property that consecutive codewords differ only in one bit position [12]. After this transformation, TGCBPM and WTGCBPM use respective number of non-matching points (NNMP) as matching criteria which are given as:

$$\begin{aligned} &NNMP_{TGCBPM,NTB}(m,n) \\ &= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\sum_{k=NTB}^{K-1} 2^{k-NTB}\{g_k^t(i,j) \oplus g_k^{t-1}(i+m,j+n)\} \end{aligned} \quad (3)$$

$$\begin{aligned} &NNMP_{WTGCBPM,NTB}(m,n) \\ &= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\sum_{k=NTB}^{K-1} g_k^t(i,j) \oplus g_k^{t-1}(i+m,j+n) \end{aligned} \quad (4)$$

where K represents the pixel-depth and NTB is the number of truncated bits, the motion block size is $N \times N$, and $-s \le m, n \le s$ is the search range. The kth most significant bit of the Gray-coded image pixel frame of time t is represented as $g_k^t(i,j)$. Note the similarity between $NNMP_{TGCBPM}$ and $NNMP_{WTGCBPM}$. Compared with the previous BPM based ME algorithms, TGCBPM and WTGCBPM based ME show significant gains in terms of ME accuracy [7][8].

B. *Multiple Candidate Motion Search*

In [11], the authors proposed multiple candidate motion searches based on two different matching criteria, i.e., $WNNMP_{2BT}(M)$ and $WNNMP_{2BT}(L)$. In general, performing many motion searches using different matching criteria simultaneously helps improving the overall performance, but their computational complexities would increase heavily with the number of matching criteria used. However, they exploited the almost identical operations between two different matching criteria and did the dual motion searches with negligible computational complexity increase. The multiple candidate motion searches based on the weighted 2BTs (MCW2BT) can be summarized as follows [11]:

1) Find two best motion vectors using two different matching criteria, respectively.
2) If two best motion vectors with the respective matching criteria are the same, declare it as the best motion vector for the current block and go to 4).
3) Calculate SADs of the two best motion vectors and declare the motion vector with less SAD as the best motion vector for the current block.
4) Go to the next current block.

It should be noted that the calculations of SADs are needed only when the two best motion vectors are different. According to the results in [11], the SAD calculations are needed about 1 out of 11 motion blocks on average which is very small. To enhance the overall ME accuracy, they also adopted the local search strategy into the MCW2BT and greatly enhanced the ME accuracy with small complexity increase.

III. PROPOSED ALGORITHMS

This section explains our proposed algorithms, which are based on Gray-coded BPMs. Since the proposed algorithms are basically extensions of the MCW2BT, in order to apply the multiple candidate motion searches, the following two conditions must be satisfied:

i) The relative local behaviors of ME results for respective matching criteria vary substantially.
ii) The operations among different matching criteria must share many identical operations.

Condition i) is for enhancing the ME accuracy. The multiple candidate motion searches take only the advantages of the respective ME results. Therefore if the local behaviors of ME results for respective matching criteria are almost the same or if a certain matching criterion based ME always outperforms the other matching criteria based ME, there is no room for improving the ME accuracy using multiple candidate motion searches. The greater the difference of ME results among different matching criteria, the better the final ME results of the multiple candidate motion searches. Condition ii) is needed for computing the different matching criteria efficiently with significantly less computations. If there are some identical operations among different matching criteria, respective matching criteria can be easily calculated using already calculated values. If this condition is not met, it is not useful for practical purposes.

To check the condition i) for the Gray-coded BPMs, we analyzed the performances of TGCBPM and WTGCBPM. Table 1 shows the average peak signal to noise ratio (PSNR) performances of TGCBPM and WTGCBPM of some CIF-size test sequences with various NTBs when the motion block size is 16×16, the search range is ±16. We can see from the Kth column of Table I that the average performance varies from sequence to sequence. For sequences of “akiyo”, “foreman”, “hall”, and “table tennis”, WTGCBPM outperforms TGCBPM and for the other sequences, TGCBPM outperforms WTGCBPM.

Unlike MCW2BT, we can also check the condition i) for TGCBPMs and WTGCBPMs with different NTBs. That is, the local behaviors of the ME results for TGCBPM and WTGCBPM with $NTB = k$ and with $NTB = l$ ($k \neq l$) are very different. From the Gth and Hth column of Table I, the average performances of TGCBPM with NTB = 5 are

TABLE I. AVERAGE SEARCH POINTS OF ALGORITHMS FOR CIF SEQUENCES WHEN THE MOTION BLOCK SIZE IS 16×16 (100-FRAME, SEARCH RANGE IS ±16)

	TGCBPM			WTGCBPM			PSNR Difference				
	A	B	C	D	E	F	G	H	I	J	K
	NTB = 6	*NTB* = 5	*NTB* = 4	*NTB* = 6	*NTB* = 5	*NTB* = 4	ΔPSNR (B-A)	ΔPSNR (C-B)	ΔPSNR (E-D)	ΔPSNR (F-E)	ΔPSNR (B-E)
football	23.41	23.55	23.26	23.39	23.68	23.68	0.14	-0.29	0.29	0.00	0.13
akiyo	42.03	42.56	42.62	41.82	42.37	42.49	0.53	0.06	0.55	0.12	-0.19
foreman	31.91	32.68	32.83	31.76	32.51	32.82	0.77	0.15	0.75	0.31	-0.17
hall	33.09	33.84	33.71	33.01	33.81	33.79	0.75	-0.13	0.80	-0.02	-0.03
bus	24.46	24.53	24.39	24.51	24.66	24.67	0.07	-0.14	0.15	0.01	0.13
tempete	27.37	27.46	27.42	27.37	27.51	27.52	0.09	-0.04	0.14	0.01	0.05
table tennis	28.11	28.37	28.30	27.93	28.18	28.29	0.26	-0.07	0.25	0.11	-0.19
children	28.89	28.68	28.46	28.91	28.93	28.90	-0.21	-0.22	0.02	-0.03	0.25
average	**29.91**	**30.21**	**30.12**	**29.84**	**30.21**	**30.27**	**0.30**	**-0.09**	**0.37**	**0.06**	**0.00**

always better, but between *NTB* = 4 and *NTB* = 5, the average performances vary slightly from sequence to sequence. However, we can observe the alternating local ME results of TGCBPM with *NTB* = 5 and *NTB* = 6 in Figure 1. Figure 1 shows the frame level PSNR results of sequence "container" using TGCBPM with *NTB* = 5 and *NTB* = 6. For the matching criterion of WTGCBPM, we can see the alternating PSNR performances very clearly in the *I*th and *H*th column of Table I.

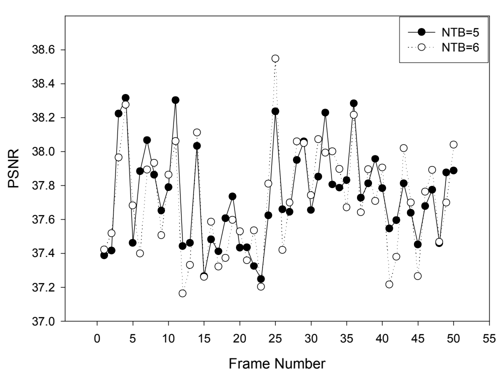

Figure 1. Frame-wise PSNR results of "container" using TGCBPM with *NTB* = 5 and *NTB* = 6 when the motion block size is 16×16 and the search range is ±16.

For checking the condition ii) for the above discussed, we define the following $NNMP_{gram,k}$ of the *k*th most significant bit as:

$$NNMP_{gram,k}(m,n) := \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} g_k^t(i,j) \oplus g_k^{t-1}(i+m, j+n) \quad (5)$$

Then, the matching error criteria of TGCBPM and WTGCBPM can be compactly represented as

$$\begin{aligned} NNMP_{TGCBPM,NTB}(m,n) &= \sum_{k=NTB}^{K-1} 2^{k-NTB} NNMP_{gram,k}(m,n) \\ NNMP_{WTGCBPM,NTB}(m,n) &= \sum_{k=NTB}^{K-1} NNMP_{gram,k}(m,n) \end{aligned} \quad (6)$$

Therefore, TGCBPM and WTGCBPM with different NTBs can be recurrently calculated as follows:

$$\begin{aligned} &NNMP_{TGCBPM,K-1}(m,n) = NNMP_{gram,K-1}(m,n) \\ &NNMP_{TGCBPM,k}(m,n) \\ &= 2 \times NNMP_{TGCBPM,k+1}(m,n) + NNMP_{gram,k}(m,n) \end{aligned} \quad (7)$$

$$\begin{aligned} &NNMP_{WTGCBPM,K-1}(m,n) = NNMP_{gram,K-1}(m,n) \\ &NNMP_{WTGCBPM,k}(m,n) \\ &= NNMP_{WTGCBPM,k+1}(m,n) + NNMP_{gram,k}(m,n) \end{aligned} \quad (8)$$

where ($0 \leq k \leq K$-2). Note that in this case, the recurrence relation is in reverse order. From the equations (6), (7) and (8), we can see that once the calculations of the values $NNMP_{gram,k}$ are finished, the final calculations of the matching criteria $NNMP_{TGCBPM,K\text{-}1}$, $NNMP_{TGCBPM,K\text{-}2}$, ···, and $NNMP_{TGCBPM,NTB}$ can be easily carried out with additional (*K*-*NTB*-1) additions and (*K*-*NTB*-1) shift operations only. In the same way, for calculations of the matching criteria $NNMP_{WTGCBPM,K\text{-}1}$, $NNMP_{WTGCBPM,K\text{-}2}$, ···, and $NNMP_{WTGCBPM,NTB}$ can be easily calculated with additional (*K*-*NTB*-1) additions only. Therefore, we can enhance the ME accuracy using multiple candidate motion searches without noticeable increase of computational complexity. Exploiting the above observations, we propose a multiple candidate motion searches based on the Gray-coded BPM (MCGCBPM) as follows:

1) Calculate the values of $NNMP_{gram,k}$ ($NTB \leq k \leq K$-1) as in (5).
2) Using (7), calculate the matching criteria of TGCBPM, i.e., $NNMP_{TGCBPM,K\text{-}1}$, $NNMP_{TGCBPM,K\text{-}2}$, ···,

TABLE II. AVERAGE PSNR AND THE NUMBER OF SAD CALCULATIONS COMPARISON OF THE ALGORITHMS ($NTB = 4$)

	1BT		2BT		TGCBPM		WTGCBPM		AM2BT		MCW2BT-LS		MCGCBPM-LS		FSBMA	
sequences	16×16	8×8	16×16	8×8	16×16	8×8	16×16	8×8	16×16	8×8	16×16	8×8	16×16	8×8	16×16	8×8
stefan	25.12	24.90	25.25	25.77	25.48	26.28	25.40	26.16	25.68 (23.95)	26.61 (13.80)	25.61 (10.65)	26.55 (12.11)	25.70 (6.63)	26.68 (7.46)	25.75	26.74
football	22.64	23.46	23.06	24.35	23.68	25.36	23.26	24.90	23.96 (56.76)	25.78 (16.13)	23.82 (11.90)	25.55 (12.18)	23.94 (7.32)	25.86 (7.40)	24.00	25.95
akiyo	41.66	35.23	42.43	41.81	42.49	42.66	42.62	42.93	42.57 (0.57)	42.61 (0.63)	42.79 (3.61)	43.07 (4.47)	42.83 (2.49)	43.44 (2.91)	42.84	43.48
foreman	31.69	30.64	31.81	31.98	32.82	33.88	32.83	34.06	32.60 (3.26)	33.66 (1.95)	33.02 (12.17)	34.11 (13.72)	33.38 (7.31)	34.99 (8.11)	33.43	35.13
mobile	23.50	23.31	23.58	23.89	23.78	24.44	23.58	24.05	23.84 (37.24)	24.6 (15.79)	23.89 (9.93)	24.71 (11.73)	23.91 (6.10)	24.79 (6.98)	23.92	24.83
hall	32.13	30.83	33.22	33.95	33.79	34.94	33.71	34.89	33.81 (2.82)	34.97 (1.29)	33.93 (11.77)	35.24 (13.68)	34.27 (7.45)	35.73 (8.46)	34.34	35.87
coastguard	29.09	28.17	29.23	29.52	29.44	30.09	29.37	30.25	29.55 (7.85)	30.36 (4.77)	29.56 (9.92)	30.41 (11.42)	29.61 (6.37)	30.64 (7.13)	29.62	30.68
container	37.57	34.81	38.12	37.82	37.90	37.69	37.98	37.84	38.16 (0.59)	38.09 (0.67)	38.24 (7.91)	38.26 (9.82)	38.24 (5.09)	38.26 (6.05)	38.33	38.42
bus	23.86	24.58	24.14	25.24	24.67	26.33	24.39	25.97	24.79 (35.79)	26.52 (11.76)	24.64 (10.84)	26.34 (11.91)	24.84 (6.72)	26.74 (7.17)	24.90	26.83
dancer	29.67	30.57	30.29	30.87	31.55	32.64	30.93	32.21	31.45 (11.37)	32.66 (4.18)	31.45 (17.55)	32.56 (17.66)	31.92 (10.74)	33.40 (10.67)	32.14	33.72
mother and daughter	37.58	35.83	39.34	38.99	39.57	39.59	39.75	40.08	39.58 (1.18)	39.67 (0.85)	39.84 (11.73)	40.36 (13.38)	40.06 (7.14)	40.86 (7.94)	40.12	41.02
tempete	27.01	26.51	27.25	27.46	27.52	27.96	27.42	27.88	27.63 (19.10)	28.16 (8.45)	27.60 (9.87)	28.18 (11.64)	27.68 (6.17)	28.33 (7.02)	27.70	28.38
table tennis	27.44	28.09	27.87	29.00	28.29	29.83	28.3	29.8	28.65 (10.61)	30.10 (4.08)	28.62 (11.79)	30.11 (12.77)	28.77 (7.16)	30.44 (7.59)	28.87	30.55
flower	25.73	26.44	25.83	26.92	25.95	27.23	25.88	27.15	25.97 (22.36)	27.30 (9.05)	26.00 (11.38)	27.37 (12.18)	26.02 (6.90)	27.43 (7.33)	26.03	27.45
children	28.05	27.75	28.32	29.64	28.90	30.55	28.46	30.16	29.16 (17.64)	30.92 (5.29)	29.08 (4.59)	30.76 (4.33)	29.19 (3.01)	31.02 (2.80)	29.24	31.10
paris	30.16	29.73	30.16	31.23	30.53	31.96	30.36	31.75	30.58 (7.33)	32.09 (2.92)	30.62 (6.91)	32.19 (8.56)	30.69 (4.27)	32.36 (5.24)	30.71	32.41
news	35.58	33.22	36.50	37.20	37.05	38.19	36.82	38.11	37.05 (3.05)	38.32 (1.38)	37.18 (5.53)	38.50 (6.64)	37.32 (3.58)	38.89 (4.16)	37.33	38.95
Average	**29.91**	**29.06**	**30.38**	**30.92**	**30.79**	**31.74**	**30.65**	**31.66**	**30.88 (15.38)**	**31.91 (6.06)**	**30.93 (9.87)**	**32.02 (11.07)**	**31.08 (6.14)**	**32.34 (6.73)**	**31.13**	**32.44**

and $NNMP_{TGCBPM,NTB}$. And using (8), calculate the matching criteria of WTGCBPM, i.e., $NNMP_{WTGCBPM,K\text{-}1}$, $NNMP_{WTGCBPM,K\text{-}2}$, ···, and $NNMP_{WTGCBPM,NTB}$.

3) Find respective candidate motion vectors according to the matching criteria calculated in 2).
4) Calculate SADs of candidate motion vectors and declare the motion vector with the least SAD as the best motion vector for the current block.
5) Go to the next current block.

To enhance the overall ME accuracy, we adopt the local search strategy into MCGCBPM. As we tested several local search algorithms with MCGCBPM, two-step search in [10] showed the best performance in terms of the PSNR performance and computational complexity. In summary, we propose the following MCGCBPM with two-step local search, namely MCGCBPM-LS:

1) Calculate the values of $NNMP_{gram,k}$ ($NTB \le k \le K\text{-}1$) as in (5).
2) Using (7), calculate the matching criteria of TGCBPM, i.e., $NNMP_{TGCBPM,K\text{-}1}$, $NNMP_{TGCBPM,K\text{-}2}$, ···, and $NNMP_{TGCBPM,NTB}$. And using (8), calculate the matching criteria of WTGCBPM, i.e., $NNMP_{WTGCBPM,K\text{-}1}$, $NNMP_{WTGCBPM,K\text{-}2}$, ···, and $NNMP_{WTGCBPM,NTB}$.
3) Find respective candidate motion vectors according to the matching criteria calculated in 2).
4) Calculate SADs of candidate motion vectors and declare the motion vector with the least SAD as the predicted motion vector.
5) Do two-step search around the predicted motion vector and find the best motion vector using SAD for the current block.
6) Go to the next current block.

(a) (b) (c) (d) (e) (f)

Figure 2. Sample results for the “foreman” sequence when the motion block size is 16 × 16 and the search range is ±16. (a) Original frame, (b) Reconstructed frame with the SAD based ME (PSNR = 30.48dB) (c) Reconstructed frame with the MCW2BT-LS ME (PSNR = 29.56dB), (d) Reconstructed frame with TGCBPM with NTB = 5 (PSNR = 29.91dB), (e) Reconstructed frame with WTGCBPM with NTB = 5 (PSNR = 29.72dB), and (f) Reconstructed frame with the proposed MCGCBPM-LS with NTB = 5 (PSNR = 30.36dB)

IV. EXPERIMENTAL RESULTS

Several experiments were carried out to compare the performances of the proposed algorithms with 1BT [3], 2BT [4], TGCBPM [7], WTGCBPM [8], AM2BT [10], MCW2BT-LS [11], and the FSBMA. The first 100 frames of 17 CIF (352 × 288) sequences (stefan, football, akiyo, foreman, mobile, hall monitor, coastguard, container, bus, dancer, mother and daughter, tempete, table tennis, flower, children, paris, and news) are used as test sequences. All the searching processes were in spiral order. For an efficient calculation of SADs, we adopted a typical PDE algorithm whose partial SAD order is 8.

A. *Performance Evaluation*

Table II shows the average PSNR results and the number of SAD calculations (if available) per motion block of the proposed MCGCBPM-LS compared with other ME algorithms when NTB = 4. When the motion block size is 16×16 and the search range is ±16, we can see that the proposed MCGCBPM -LS outperforms 1BT and 2BT based ME by 1.17dB and 0.70dB, respectively. And since the proposed MCGCBPM-LS takes only the advantages of TGCBPM and WTGCBPM, we can expect that the performance of the proposed algorithm will always be better than TGCBPM and WTGCBPM based ME, and the table shows that it is the case. Compared with AM2BT and MCW2BT-LS, the proposed MCGCBPM-LS is better in terms of both the PSNR performance and the computational complexity. To be specific, for PSNR performance, the proposed algorithm is better than AM2BT and MCW2BT-LS by 0.20dB and 0.15dB, respectively. For computational complexity in terms of SAD computations, the proposed MCGCBPM-LS requires 39.92% of AM2BT and 62.21% of MCW2BT-LS. Compared with the performance with FSBMA, FSBMA outperforms the proposed MCGCBPM-LS by 0.05dB, which is very small. To be specific, for sequences “akiyo”, “mobile”, “coastguard”, “tempete”, “flower”, “paris”, and “news”, the PSNR performance differences are within 0.03dB.

When the motion block size is 8×8 and the search range is ±8, the PSNR performance differences between the proposed MCGCBPM-LS and 1BT, 2BT, TCGBPM, WTGCBPM, AM2BT, and MCW2BT-LS are more than twice compared with the results when the motion block size is 16×16. For computational complexity, the proposed MCGCBPM-LS requires 60.79% of MCW2BT-LS in terms of SAD computations. In contrast, the proposed MCGCBPM -LS requires 11.06% more SAD computations compared with AM2BT. However, since the PSNR improvement of the proposed MCGCBPM-LS over AM2BT is about 0.43dB, we presume that this slight computational complexity increase is tolerable.

B. *Visual Quality Evaluation*

To compare the visual quality of the proposed algorithm with other algorithms (TGCBPM, WTGCBPM, MCW2BT-LS, and FSBMA), the reconstructed frames of “foreman” are given in Figure 2. The motion block size is 16×16, the search range is ±16, and NTB = 5. As can be seen from the figures, many bad motion vectors are observed in the reconstructed frames of TGCBPM and WTGCBPM resulting in annoying visual quality. For MCW2BT-LS, some bad motion vectors are seen in the reconstructed frames. However, in case of the proposed MCGCBPM-LS, bad motion vectors are reduced significantly and its visual appearance is much more pleasing and almost the same as that of the FSBMA, which can be expected from the PSNR results.

V. CONCLUSIONS

Low complexity ME algorithms based on the Gray-coded BPM are proposed in this paper. By exploiting almost identical operations among similar but different matching error criteria, we can efficiently determine the respective candidate motion vectors. Moreover, adopting multiple candidate ME strategies into those candidate motion vectors and two-step local searches around the best candidate motion vector enhance the motion estimation accuracy substantially with relatively small computational complexity increase. Experiments were carried out for comparing the performance of the proposed algorithms with other BPM based motion estimation algorithms, and FSBMA as well. The proposed MCGCBPM -LS outperforms all the other BPM based MEs with negligible complexity increase. The PSNR difference between the proposed MCGCBPM-LS and FSBMA is within 0.05dB on average for $NTB = 4$.

REFERENCES

[1] Information Technology - Coding of Audio Visual Objects - Part 2: Visual, JTC1/SC29/WG11, ISO/IEC 14496-2 (MPEG-4 Visual), 2002.

[2] Advanced Video Coding for Generic Audiovisual Services, ITU-T Recommendation H.264, May 2005.

[3] B. Natarajan, V. Bhaskaran, and K. Konstantinides, “Low-Complexity Block-based Motion Estimation via One-Bit Transforms,” IEEE Trans. Circuits and Systems for Video Tech., vol. 7, no. 5, Aug. 1997, pp. 702-706.

[4] S. Erturk, “Multiplication-Free One-Bit Transform for Low-Complexity Block-Based Motion Estimation,” IEEE Signal Processing Letters, vol. 14, no. 2, Feb. 2007, pp. 109-112.

[5] O. Urhan and S. Erturk, “Constrained One-Bit Transform for Low Complexity Block Motion Estimation,” IEEE Trans. Circuits and Systems for Video Tech., Apr. 2007, vol. 17, no. 4, pp. 478-482.

[6] A. Erturk and S.Erturk, “Two-Bit Transform for Binary Block Motion Estimation,” IEEE Trans. Circuits and Systems for Video Tech., vol. 15, no. 7, Jul. 2005, pp. 938-946.

[7] A. Celebi, O. Akbulut, O. Urhan, and S. Erturk, “Truncated Gray-Coded Bit-Plane Matching Based Motion Estimation Method and Its Hardware Architecture,” IEEE Trans. Consumer Electron., vol. 55, no. 3, Aug. 2009. pp. 1530-1536.

[8] A. Celebi, H. Lee, and S. Erturk, “Bit Plane Matching Based Variable Block Size Motion Estimation Method and Its Hardware Architecture,” IEEE Trans. Consumer Electron., vol. 56, no. 3, Aug. 2010, pp. 1625-1633.

[9] C. Choi and J. Jeong, “Enhanced Two-bit Transform based Motion Estimation via Extension of Matching Criterion,” IEEE Trans. Consumer Electron, vol. 56, no. 3, Aug. 2010, pp. 1883-1889.

[10] B. Demir and S. Erturk, “Block Motion Estimation Using Adaptive Modified Two-Bit Transform,” IET Image Processing, vol. 1, no. 2, Jun. 2007, pp. 215-222.

[11] C. Choi and J. Jeong, “Low complexity Weighted Two-bit Transforms based multiple candidate Motion Estimation,” IEEE Trans. Consumer Electron, vol. 57, no. 4, Nov. 2011, pp. 1837-1842.

[12] S. Ko, S. Lee, S. Jeon, and E. Kang, “Fast Digital Image Stabilizer based on Gray-coded Bit-plane Matching,” IEEE Trans. Consumer Electron., vol. 45, no. 3, Aug. 1999, pp. 598-603.

 ISBN: 978-1-61208-452-7