True Motion Estimation Based on Reliable Motion Decision

Tien-Ying Kuo, Cheng-Hong Hsieh, Yi-Chung Lo, Jian-Hua Wang
Department of Electrical Engineering, National Taipei University of Technology, Taiwan, R.O.C.
Email:tykuo@ntut.edu.tw,{chhsieh , yclo , jhwang}@image.ee.ntut.edu.tw

Abstract— **This paper presents a novel technique to estimate true motion vectors (TMVs) during video encoding. The proposed method designs a system unit to collect motion vectors' (MVs') characteristic in the video encoder. MVs are then classified into reliable and unreliable MVs based on the collected early-stop distribution of each block. The reliable MVs are directly assigned as TMVs, whereas the unreliable MVs must go through the refinement processing for TMV conversion. This algorithm can be easily integrated into the existing video encoders. The experiment result shows that proposed method is superior to the literature works in terms of either performance or computation complexity.**

I. INTRODUCTION

When video applications encounter a limited available channel bandwidth, they usually sacrifice the video frame rate for transmitting the video. The corresponding decoder can then create some interpolated frames to restore to the original video frame rate. This technique to restore the frame rate is called the frame rate up-conversion (FRUC) algorithm. FRUC helps the video playback smoothly and reduces the flickering artifacts [1]. Another important application of FRUC is to apply to video sources conversion between two different video frequencies, such as 25Hz PAL and 30Hz NTSC videos.

Motion-compensated frame interpolation (MCFI) [2] is an important core in FRUC for video frame interpolation. The quality of the interpolated frame with MCFI will strongly rely on obtaining accurate object motion trajectories between frames, i.e., providing the accurate true motion vectors (TMVs). However, the goal of the conventional motion estimation (ME) used in the video compression aims at increasing the coding efficiency instead of finding the TMVs. Therefore, it is important to design an algorithm specially for estimating TMV to improve the MCFI [2]-[7] quality. Furthermore, the accuracy of TMV can affect other kinds of video applications, such as error concealment [9], object tracking [8], [9], shot change detection [10], image stabilization system [11] and slow motion replay.

The cost function used in the conventional ME is commonly based on sum of absolute difference (SAD) or sum of square error (SSE) to evaluate the MV candidates during block match processing. The cost function is pretty sensitive to the noise and shadow variation. Therefore, it is very difficult find out the TMV merely using the conventional ME. Choi et al. [3] proposed a bi-directional motion estimation algorithm, and smoothed the MV in the spatio-temporal domain. This algorithm makes the obtained MVs consistent locally, but can be still affected by the noises and shadow variation. Ha et al. [4] adopted the overlapped block-based motion estimation (OBME) and utilized motion vector median filter (VMF) [12] as the post-processing to obtain the TMV. The OBME basically adopts an enlarged estimation block for matching the block with more correct motion trajectory in contrast to a small-sized block. However, OBME is high computational complexity involved. To trade off the performance and complexity, Zhai et al. [2] and Yang et al. [5] utilized the conventional ME and its correspondent SAD information to classify MVs as the reliable and the unreliable MVs, and then only re-estimated the unreliable MVs with OBME which block size are further modified from 16×16 to 12×12 to improve the computational complexity. The post processing VMF [12] is then applied to smooth motion fields. The complexity of this method is still much higher than the conventional ME with block size 8×8.

In contrast to the block match (BMA) based algorithms mentioned above, the 3D recursive search block matching (3DRS) [6] is very effective in finding TMV, which selects the best MV from candidate MVs and updates the candidate MVs iteratively. The criterion of iteration convergence controls the performance and complexity. Tai et al. [7] provided a multi-pass ME algorithm with larger block size to detect the motion vectors within the objects themselves, while with smaller block size to detect the motion vectors along the objects' boundary. Both the above algorithms are iteration-based with high complexity involved, and their design is different from the conventional ME and thus they are hard to be integrated into the existing embedded system or integrated circuit of the video encoder ME unit.

This study proposes a novel framework to obtain the true motion trajectories during video encoding. First, our MV analyzer estimates MVs and records the side information in the ME processing, including SAD and early-stop information for each block. Then our MV classifier categorizes the MVs into reliable and unreliable MVs. The reliable MVs are directly set as TMVs. However, the unreliable MVs are further classified to one of the unmatched, the unrelated, and the uncertain MVs. Finally, the different refinement processing is applied to the unmatched, the unrelated, and the uncertain MVs. The unmatched MVs are post-processed by median filter with local MVs. The unrelated MVs and the uncertain MVs are estimated again with the smooth constraint. Not only this proposed algorithm can find out the accurate TMV with low complexity, but also it can be easily integrated into the most of popular block-based video encoders, such as MPEG-x and H.26x, since our method mainly relies on the side information of the existing ME.

The rest of this paper is organized as follows. Section 2 describes our proposed method. Section 3 shows the experiment result of our algorithm. Finally, the conclusion is drawn in the Section 4.

II. PROPOSED METHOD

Figure 1 illustrates our proposed algorithm which is embedded with a conventional ME unit. Our method is composed of three

This work was supported by the National Science Council under Grants: NSC 101-2221-E-027-078-MY2.

stages: MVs analyzer, MVs classifier and MVs refiner. Giving an target image block C of current video frame F_t, MVs analyzer first estimates the best motion vector $MV_{C,t}$, and calculates its corresponding side information based on the current frame F_t, the previous frame $F_{t\text{-}1}$ frame, the early-stop histogram $Hist_{C,t}$ and $SAD_{C,t}$. Based on $Hist_{C,t}$ and $SAD_{C,t}$, we can classifies $MV_{C,t}$ into the reliable MV ($RMV_{C,t}$), the unmatched MV ($UMMV_{C,t}$), the unrelated MV ($URMV_{C,t}$), or the uncertain MV ($UCMV_{C,t}$). Finally, our MV refiner processes the unmatched, the unrelated, the uncertain MVs to TMVs by the help of F_t, $F_{t\text{-}1}$, TMV_t, and $TVM_{t\text{-}1}$.

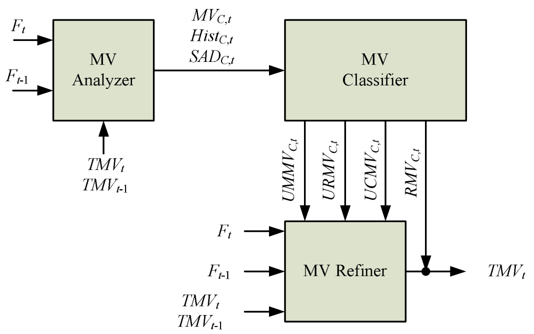


Fig. 1 The proposed framework

A. MV Analyzer

This stage is embedded with a conventional BMA-based motion search unit with the match block size $N\times N$ pixels and the search range $(2M+1)\times(2M+1)$ pixels. The block diagram of our MV analyzer is shown in Fig 2. First, the median motion vector predictor produces a predicted motion vector $PMV_{C,t}$ from neighboring vectors of previously estimated TMV_t. Then, the BMA-based ME unit estimates the best MV within the search range and performs with a smooth constraint. The side information, $Hist_{C,t}$ and $SAD_{C,t}$ during the ME processing will be calculated and recorded for each block.

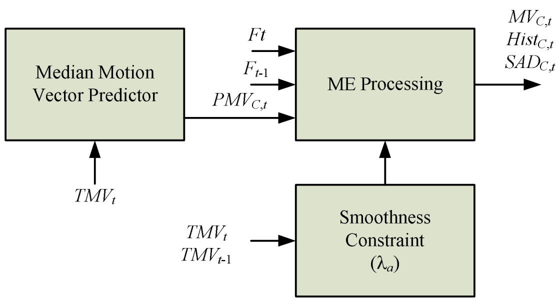


Fig. 2 MV analyzer

The smooth constraint exploits the spatial consistency and temporal smoothness in the ME processing to find out the MV close as true motion to reduce the influence from the noises and shadow variation. The ME cost function $Cost(.)$ we use is shown as (1).

$$Cost_{C,t}(p,q) = SAD_{C,t}(p,q) + \lambda_a \times SC_{C,t}(p,q), -M < p,q < M \quad (1)$$

$$SAD_{C,t}(p,q) = \sum_{i=i_0}^{i_0+N-1} \sum_{j=j_0}^{j_0+N-1} \left| F_{C,t}(i,j) - F_{C,t-1}(i-p, j-q) \right| \quad (2)$$

$$SC_{C,t}(p,q) = \sum_{i=1}^{5} \left[(p - mv_{x_i})^2 + (q - mv_{y_i})^2 \right] \quad (3)$$

where (p, q) indicate the location of the block's center pixel in the previous frame $F_{C,t\text{-}1}$; λa is an parameter to join the function SAD(.) and SC(.), which is set as 0.25 to provide satisfactory results in our test; SAD(.) represents the sum of absolution difference; SC(.) represents the smooth constraint. The smooth constraint takes into account the relationship of target block and the neighboring blocks as well as the previously corresponding block (expressed as index i in (3)), as shown in Fig. 3.

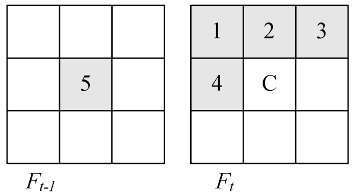


Fig. 3 The reference blocks of smooth constraint

The approach of adding a smooth constraint (SC) in ME cost function (i.e., smoothing in advance during ME) is better than the ME with the post processing VMF [12] (i.e., smoothing afterwards after ME). As shown in Fig. 4 about the comparison of SC versus VMF, we can observe that the method SC is more accurate than VMF.

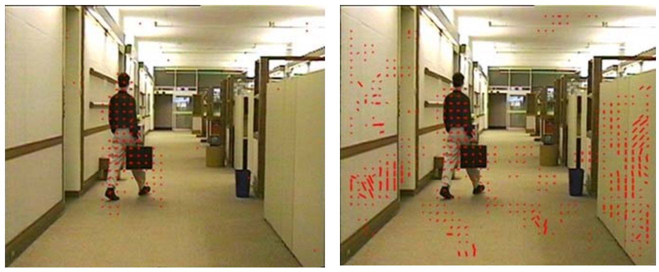

(a) TMVs by SC processing (b) TMVs by VMF processing

Fig. 4 SC vs. VMF

B. MV Classfier

There are four types of blocks resulting in different block match conditions in the conventional ME as described in [8]:

Unrelated block: The target block is with homogeneous intensity. There are many small SAD among the search area.

Matched block: The target block contains heterogeneous texture. The obtained MV can be considered as TMV with high confidence. The minimum SAD of this block among the search area is small and unique.

Unmatched block: The target block cannot find any similar block from previous frame; for example, the blocks belongs to the fast motion objects, occlusion area, shot change frame, and so on.

Uncertain block: The target block is with repeated pattern. There are many small SAD among the search area and occurs periodically.

According to the above observation, this stage will distinguish the MVs into four types based on their corresponding side information, $Hist_{C,t}$ and $SAD_{C,t}$. which are RMV, UMMV, URMV, and UCMV. Figure 5 shows the decision flow of the MV classifier. First, the unmatched block has an obvious feature which exists a high SAD value in the block. Therefore, by comparing SAD with a threshold TH_a, it is able to identify UMMV with TH_a set as 200 to get satisfactory results in our test.

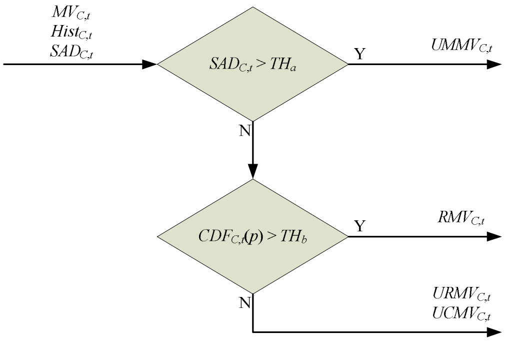
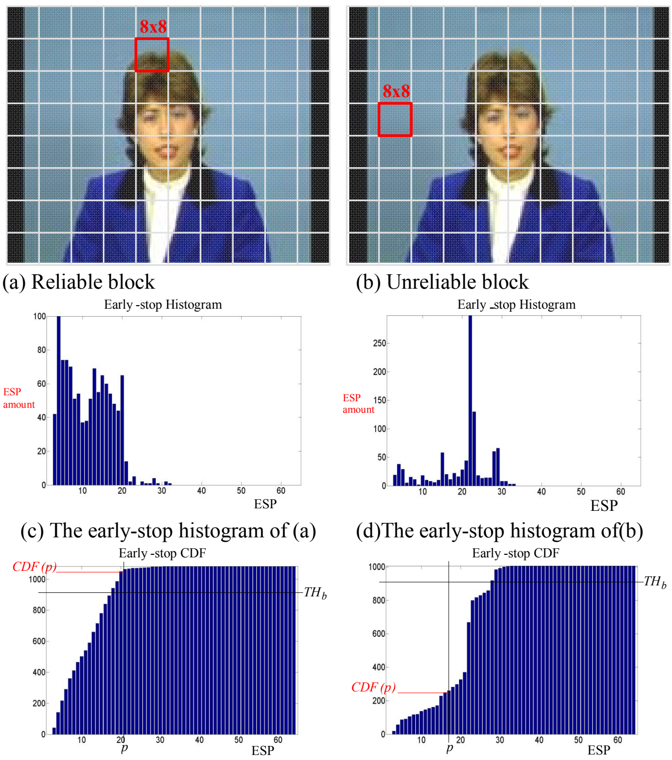


Fig. 5 MV Classifier

After identifying UMMV, MVs classifier further classify the reliable MV based on the side information, and the early-stop histogram, $Hist_{C,t}$. Figure 6 illustrates the concept why $Hist_{C,t}$ can be used in determining the MV reliability. In Fig. 6(b), a block with homogenous background will result in the URMV, since the block matching could easily be sensitive to illumination variation. As shown in Fig. 6(d), the block usually accumulates the larger histogram value in the larger Early stop point (ESP) index, since most search points give similar and small SAD values and the early stop will then not be very efficient. On the contrary, the block in Fig. 6(a) consists of the object and background boundary, and the results of the motion matching is more robust and reliable. Therefore, only few of search points have very small SAD values and the early stop will work earlier and often here, which results in the larger histogram value in the smaller ESP index as shown in Fig. 6(c).



(a) Reliable block
(b) Unreliable block
(c) The early-stop histogram of (a)
(d)The early-stop histogram of(b)
(e) The early-stop CDF of (c)
(f) The early-stop CDF of (d)

Fig. 6 The early-stop histogram and CDF

The ESP histogram examples corresponding to Figs. 6(a) and (b) are illustrated in Figs. 6(c) and (d) with the case of the block size 8×8 and search range 33×33, i.e., N=8 and M=16. We observe that the ESP histogram distribution of a reliable block MV is much concentrated on the left part in the range of smaller ESP index in Fig. 6(c) in contrast to the unreliable block MV in Fig. 6(d), which has sparse distribution when ESP index is small.

We plot the cumulative distribution function (CDF) of Figs. 6(c) and (d) in Figs. 6(e) and (f). Based on the CDF distribution, we propose a decision rule to check the reliability of the block MV, i.e., it is true MV or not, by (4).

$$MV_{C,t} = \begin{cases} RMV_{C,t} & ,CDF(p) > TH_b \\ URMV_{C,t} \text{ or } UCMV_{C,t} & ,CDF(p) \le TH_b \end{cases} \tag{4}$$

where CDF(p) denotes the number of accumulation of the ESP histogram from ESP index 0 to p. THb is a pre-set threshold with value of α(2M+1)×(2M+1). In this work, α is set as 0.85, and p is set as the middle point of ESP index, i.e., 0.5× (N×N).

C. *MV refiner*

Figure 7 shows the block diagram of MVs refiner in detail. For UMMV, the stage tries to refine it by exploiting the MVs consistency with the neighboring MVs since the block type is unmatched and the ME does not work at all in such the case of the block containing object occlusion. VMF [12] is suitable and applied here to refine UMMVs. As for URMVs and UCMV, they can be refined by the smooth constraint owing to the discussion of Fig. 4. In this stage, the smooth constraint acts as stronger consistency constraint to neighboring MV's than the case in the MV analyzer. Therefore, smooth constraint (1) can be reused here as (5) but with much larger value of λb (λb =12) compared to λa in (1).

$$Cost_{C,t}(m,n) = SAD_{C,t}(m,n) + \lambda_b \times SC_{C,t}(m,n) \tag{5}$$

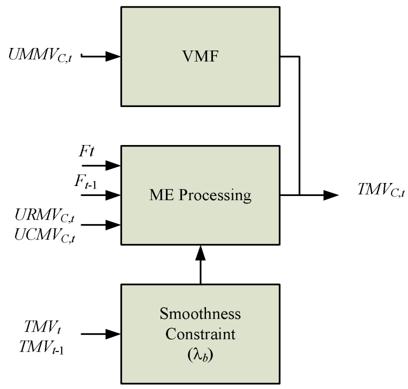


Fig. 7 MV refiner

III. Experiments Result

The video sequences in test are Akiyo, Bream, Carphone, Football, Foreman, Mother Daughter, News, Silent, and Stefan in CIF format. Our test-bed is implemented in C and run on a PC with an Intel® CoreTM 2 Duo 2.33GHz CPU and 2GBytes ram. First, the frame rate of the test sequences is down-sampled either to 10 fps or 15fps from original 30 fps. The down-sampled video sequences are then used to estimate the TMVs with three algorithms, including 3DRS [6], Zhai's [2], and our proposed algorithm. These TMVS are applied to the down-sampled video to restore the frame rate back to 30 fps using bi-directional motion compensated interpolation (BMCI) [3] algorithm. Table I shows the PSNR evaluation for each method by comparing the up-converted video from 10 fps with the original video at 30 fps.

TABLE I
AVERAGE PSNR FOR BMCI FRAMES FROM 10FPS

Video sequence	3DRS [6]	Zhai [2]	proposed
Akiyo	36.94	37.44	**37.65**
Bream	23.78	23.98	**24.19**
Carphone	28.13	28.24	**28.49**
Football	20.18	20.11	**20.32**
Foreman	26.01	26.02	**26.29**
Mother daughter	34.57	34.73	**35.01**
News	29.85	29.96	**30.26**
Silent	30.03	29.97	**30.25**
Stefan	18.86	18.71	**18.93**
Average	27.59	27.68	**27.93**

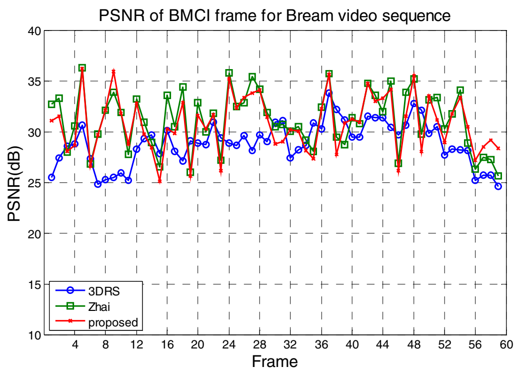
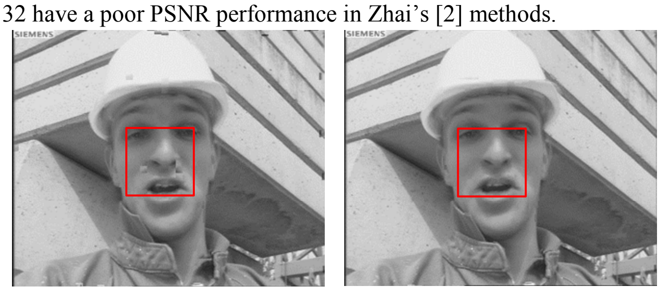


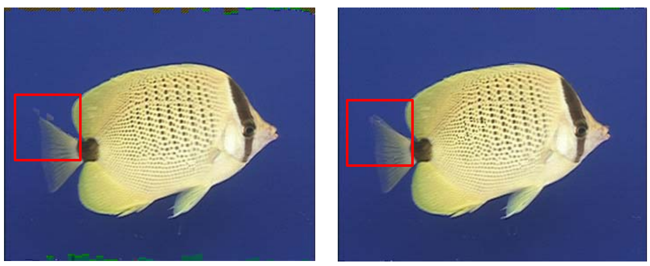
Fig. 8 PSNR of BMCI frame for Bream video sequence

From Table I and Figs. 8 to 10.The performance of our proposed method is better than 3DRS [6] and Zhai's [2] methods The particular frames are deliberately chosen here because frames 94 and 32 have a poor PSNR performance in Zhai's [2] methods.



(a) BMCI frame 94 based on Zhai's methods (b) BMCI frame 94 based on proposed method

Fig. 9 BMCI frame comparison in Foreman



(a) BMCI frame 32 based on Zhai's methods (b) BMCI frame 32 based on proposed method

Fig. 10 BMCI frame comparison in Bream

Since TMV describes the object motion trajectories, BMCI frame's quality will highly rely on TMV accuracy. Table II shows the speed comparisons among three algorithms with two test cases of the up-conversion from 10fps to 30fps, and 15fps to 30fps. This table proves that our proposed method is faster than the other two literature methods.

TABLE II
AVERAGE ELAPSED TIME (MS/FRAME)

Method	10Hz	15Hz
3DRS [6]	504	450
Zhai [2]	1557	1459
Proposed	756	900

IV. CONCLUSION

This study proposes a novel TMV estimation algorithm. First, the proposed algorithm uses the side information of ME processing to classify the MV into TMVs, URMVs, UMMVs, and UCMVs. Then, two different post-processing methods are applied to refine the URMVs, UMMVs, and UCMV into TMVs. The experiment's result shows that the proposed algorithm is much better than the literature works in terms of PSNR and complexity, especially for lower frame rate FRUC application, such as from 10fps to 30fps.

REFERENCES

[1] E. Petajan, "The HDTV Grand Alliance System," in Proc. IEEE Communications Magazine, vol. 34(6), pp. 126-132, Jun.1996.

[2] J. Zhai, K. Yu, J. Li, and S. Li, "A low complexity motion compensated frame interpolation method," in Proc.IEEE ISCAS, May 2005, pp. 23–26.

[3] B. T. Choi, S.-H. Lee, and S. J. Ko, "New frame rate up-conversion using bi-directional motion estimation," IEEE Trans. Consum. Electron., Vol. 46, no. 3, pp. 603–609, Aug. 2000.

[4] T. Ha, S. Lee and J. Kim, "Motion Compensated Frame Interpolation by new Block-based Motion Estimation Algorithm," IEEE Transactions on Consumer Electronics, Vol. 50, Issue 2, pp.752-759, May 2004.

[5] Y. T. Yang, Y. S. Tung and J. L. Wu, "Quality enhancement of frame rate up-converted video by adaptive frame skip and reliable motion extraction," IEEE Trans. Circuits Syst. Video Technol., Vol. 17, no. 12, pp.1700–1713, Dec. 2007.

[6] G. D. Haan, P.W.A.C. Biezen, H. Huijgen, and O.A. Ojo, "True-Motion Estimation with 3-D Recursive Search Block Matching," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 3, NO.5, OCTOBER 1993.

[7] S. C. Tai, Y. R. Chen, Z. B. Huang, and C. C. Wang, "A multi-pass true motion estimation scheme with motion vector propagation for frame rate up-conversion applications," Journal of Display Technol., Vol. 4, no. 2, pp. 188–197, June 2008.

[8] Z. Zhang, D. Liu and H. Wu, "Frame rate up-conversion based on matching classification and dominant object tracking," IEEE International Conference on Network Infrastructure and Digital Content, pp.981-985, 2009.

[9] H. Yuan, Y. Chang and Y. Ma, "A Moving Objects Extraction Method Based On True Motion Information," International Conference on Visual Information Engineering, pp. 638-642, 2008.

[10] Y. Hu, B. Han, G. Wang and X. Lin, "Enhanced Shot Change Detection using Motion Features for Soccer Video Analysis," IEEE International Conference on Multimedia and Expo, pp.1555-1558,2-5 July 2007.

[11] T. Y. Kuo, C. H. Wang, "Fast Local Motion Estimation and Robust Global Motion Decision for Digital Image Stabilization," IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP08), pp.442-444, Aug. 2008.

[12] J. Astola, P. Haavisto, Y. Neuvo, "Vector median Filters," Proceedings of the IEEE, Vol. 78, Issue 4, pp. 678-689,April 1990.