Advanced Motion Vector Smoothing For Frame Rate Up-Conversion

Yonghoon Kim and Jechang Jeong

Abstract— In this paper, we propose a new motion vector smoothing algorithm to be used for motion-compensated frame rate up conversion (FRUC). The FRUC techniques are used to enhance the visual quality of the low frame rate video on display. Motion estimation process searches a motion vector which refers the relative block position where it has the minimum sum of difference (SAD) between the reference and target block. By using the motion vectors obtained from motion estimation process, the new frames can be interpolated. Because of plane region and repeated pattern of the background, motion estimation sometimes produces outliers. To detect and correct the outliers of motion vectors, motion vector smoothing process is utilized. In identifying the motion vector outliers, the proposed algorithm uses distance of neighboring motion vectors as weights for producing corrected motion vector. The experimental result shows that the proposed algorithm gives the best objective and subjective results.

Keywords—Frame rate up-conversion, interpolation, motion estimation

I. Introduction

FRAME rate up-conversion refers to the technique that increase frame rate of the video from the one with a lower frame rate through the process of producing new frames and inserting them into the original video. FRUC has been used for the conversion between two display formats with different frame rates, or to remove the temporal redundancy in video coding. Recently, FRUC has become a new area of applications to the reduction of motion blur of the hold-type displays such as liquid crystal display (LCD) [1], [2]. As the motion blur of the moving image on the LCD can be greatly reduced, major display manufacturers increase the frame rate of the TV from 60 Hz to 120 or 240 Hz.

Normally, motion compensated FRUC conducts two processes for interpolating a new frame. First part is motion estimation (ME). The ME process searches motion vectors by calculating the displacement of moving objects between two or more consecutives frames. Second one motion compensated interpolation (MCI) [3] which is a process to make an interpolated frame with motion vectors obtained from ME process. Image quality of the FRUC obviously depend on the accuracy of the motion vector, therefore motion compensation in FRUC needs to find the true motion trajectory rather than the one with minimum residual energy.

Until now, several ME algorithms have been introduced and the most popular method is block matching algorithm (BMA) because of its superior scalability [4-6] and easy to implement in software and hardware [7], [8]. In FRUC, there are two motion estimation methods, bidirectional and unidirectional ME (UME). The bidirectional ME (BME) is an efficient method to reduce hoes and occluded region. Many BME algorithms are proposed such as extension of original BME [9], overlapped BME methods [10-12], and multi frame based BME [13, 14]. To overcome the weakness of the UME method, several algorithm try to use both forward and backward motion vector to better motion vector estimation [4, 15] and other algorithm using both BME and UME [16].

Either BME or UME algorithm produces outliers which mean incorrect motion vector and many algorithm motion vector smoothing method to correct the outliers by using the information relating to adjacent blocks in the MV field [17]. It is generally used for post-MV processing after the ME process. Many existing motion vector smoothing (MVS) algorithms successfully detect and correct the outliers. However, previous work reported that they still suffer from false MV correction and residual error after correction

In this paper, we presented advanced MVS method utilized the correlation between neighboring MVs to adaptively generate the MV.

II. Proposed algorithm

A. *Bidirectional motion estimation*

Conventional BME searches a motion vector by using temporal symmetry between the corresponding blocks in the previous and current frames as shown in Fig. 1. When finding the best matching block, the BME uses the sum of bidirectional absolute differences (SBAD) which is given as follows:

$$SBAD(dx,dy) = \sum_{x \in W_x} \sum_{x \in W_y} \left| f_{n-1}(x-dx, y-dy) - f_n(x+dx, y+dy) \right| \tag{1}$$
$$v = \arg \min_{(dx,dy)\in S} \{SBAD(dx,dy)\}$$

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT and future Planning(NRF-2015R1A2A2A01006004)

Y. Kim is with the Department of Electronics and Computer Engineering, Hanyang University, Haengdangdong, Sungdonggu, Seoul, South Korea (e-mail: hoonykim85@gmail.com).

J. Jeong is with the Department of Electronics and Computer Engineering, Hanyang University, Haengdangdong, Sungdonggu, Seoul, South Korea (corresponding author, e-mail: jjeong@hanyang.ac.kr).

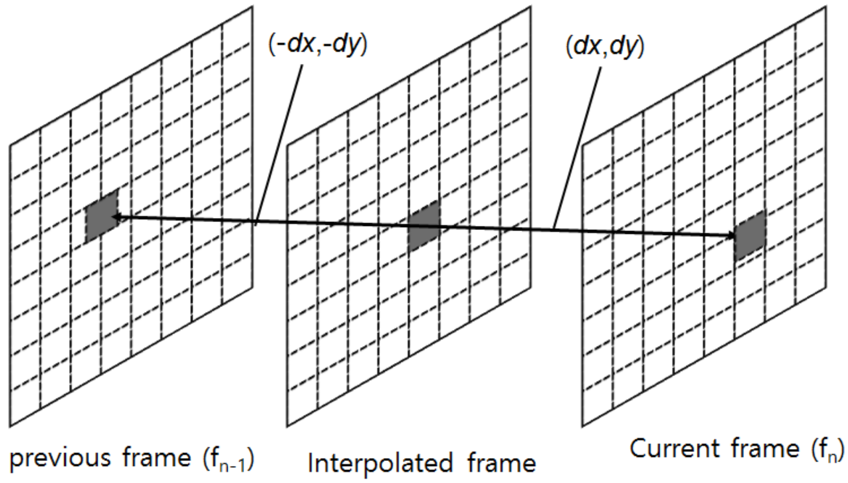


Fig. 1 Bilateral motion estimation

where (dx,dy) denotes the candidate motion vector, and f_{n-1} and f_n are the previous and current frames. W_x and W_y denote the horizontal and vertical positions of a block respectively, and S represents the search window. From eq. (1), the final motion vector is selected when the candidate motion vector has the minimum SBAD value in the search window.

v_1	v_2	v_3
v_4	v_c	v_5
v_6	v_7	v_8

Fig. 2 3x3 motion vector window

B. *Proposed Advanced Motion Vector Smoothing (AMVS)*

After obtain the motion vectors using bidirectional ME, the proposed algorithm checks the motion vector reliability using the neighboring motion vectors to find the outliers. The proposed AMVS process uses a 3 x 3 window as shown in fig. 2 [15]. The outliers are detected by calculating the mean (v_m), mean of the distance between each motion vector and v_m (ψ_m) and the distance between v_m and target MV (ψ_0). These values are given as follows:

$$\begin{aligned} v_m &= \frac{1}{8}\sum_{i=0}^{8} v_i, \\ \psi_m &= \frac{1}{8}\sum_{i=1}^{8} \|v_i - v_m\|, \\ \psi_0 &= \|v_c - v_m\|. \end{aligned} \tag{2}$$

If ψ_0 is larger than ψ_m, v_c is marked as outlier because it can be considered it lies far from other MV group. If v_c is marked as outlier, than we need to correct the MV.

The proposed algorithm considers the reliability of the eight neighboring MV to produce more accurate MV using given information. Previous algorithms use median filter or just mean of reliable neighboring motion vectors. However, the proposed algorithm produces the corrected MV adaptively by using the weight w based on distance from the v_m of each MV which is given as:

$$w_i = \frac{1}{(v_i - v_m)^2}, \tag{3}$$

where w_i denotes weight for ith MV. Using these values the final MV ($v_c^{'}$) is calculated as follow

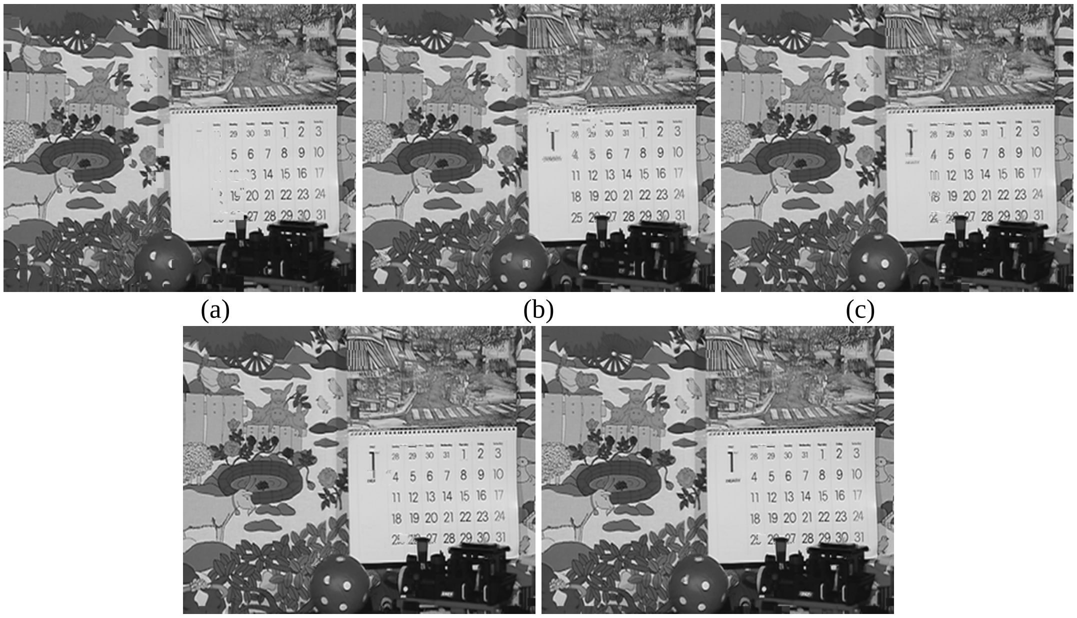
$$v_c^{'} = \frac{\sum_{i=1}^{8} w_i \times v_i \times m_i}{\sum_{i=1}^{8} w_i \times m_i}, \tag{4}$$

where m_i denotes a mask which is 0 when v_i is outlier and 1 when v_i is inlier.

This calculation process is adopted vertical and horizontal MV estimation, respectively. The proposed algorithm considers the amount of reliability among the inlier motion vectors by using weight based MVS. Therefore the proposed method can improve the accuracy of the estimation process.

III. Experimental Result

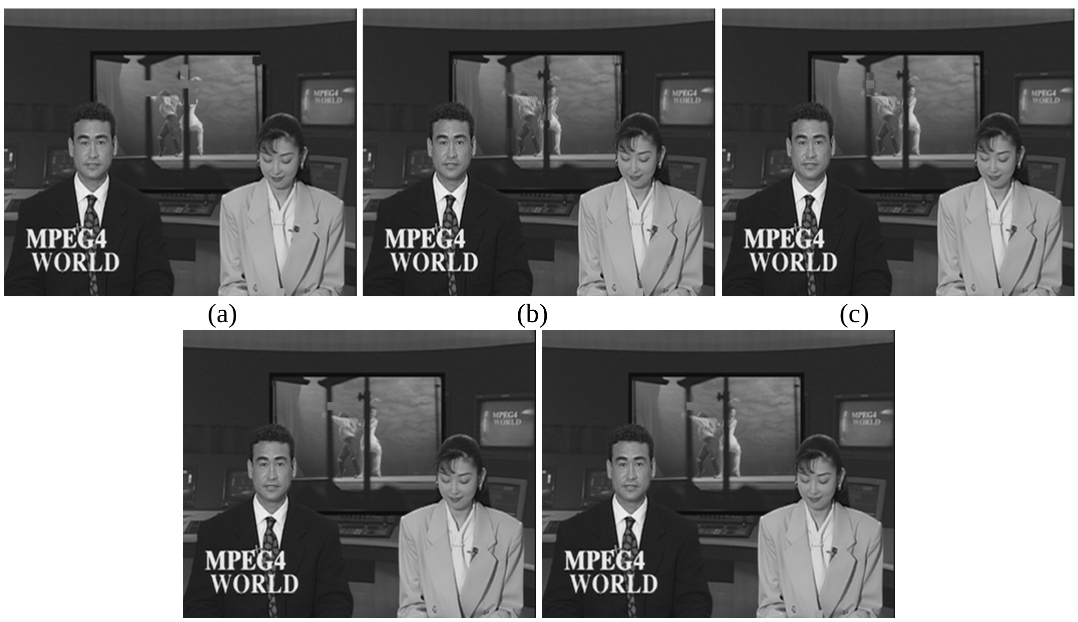
In the experiments, we set the block size and overlapped block size to 8 x 8 and 12 x 12 and the search range was is set as 16 pixels. We have conducted simulation experiments on 7 standard image sequences of common intermediate format (CIF: 352 x 288) and 30 Hz frame rate. For the test sequences, we remove the odd frames and interpolate new frames from the



(a) (b) (c)

(d) (e)

Fig. 3 Interpolated frame of *Mobile*, obtained using (a) BME_only, (b) BME with [11], (c) BME with [13], (d) BME with [9], and (e) BME with proposed algorithm



(a) (b) (c)

(d) (e)

Fig. 4 Interpolated frame of *News*, obtained using (a) BME_only, (b) BME with [11], (c) BME with [13], (d) BME with [9], and (e) BME with proposed algorithm

even frames using the FRUC algorithms. A purposed of the experiment is to compare the performance of the MVS algorithms; therefore every algorithm is tested on same condition. To evaluate the performance of the proposed algorithm, it is compared with bidirectional MV algorithm without MVS, bidirectional MV with MVS algorithm used in [11], [15], and [9]. MVS method used in [11] and [15] are based on the median filtering method and [9] adopts the MV averaging. We evaluated the objective image quality of the interpolated frames from the proposed and existing methods in terms of the peak signal-to-noise ratio (PSNR) which is given as:

$$PSNR = 10\log_{10}\frac{255\times255}{\frac{1}{W\times H}\sum_{i=0}^{H-1}\sum_{j=0}^{W-1}(f_n(i,j)-\hat{f}_n(i,j))^2} \tag{5}$$

where W and H denote width and height of the frame, respectively. f_n and $\hat{f}_n$ denote the original frame and interpolated frame, respectively.

Table I shows the PSNR results and the highest values are marked as bold. From the results, the proposed MVS algorithm

TABLE I
PNSR PERFORMANCE COMPARISON

	No_MVS	[15]	[11]	[9]	Proposed
Forman	26.23	30.29	29.89	30.35	**30.74**
Mobile	23.22	25.56	26.11	26.82	**27.54**
football	23.08	22.81	23.37	**23.50**	23.48
News	30.76	33.70	34.14	34.47	**34.86**
Stefan	23.29	24.40	24.38	24.89	**24.96**
Bus	20.17	21.03	21.41	21.98	**22.03**
CoastGuard	30.38	31.96	32.91	33.24	**33.71**
Average	25.30	27.11	27.46	27.89	**28.19**

shows the best objective performance in terms of PSNR. The proposed algorithm improved the average PSNR by 1.0 dB, 0.7 dB, and 0.3 dB comparing to method used in [15] [11], and [9], respectively. Except the football sequence, proposed algorithm has the best objective performance.

Fig.3 and 4 show the subjective performance comparison. In Fig.3 (a), the characters in the calendar area are severely corrupted when MVS algorithm is not applied. The MVS algorithms significantly improve the subjective results however there are still outlier MV. The proposed algorithm produces most clear image than other algorithms. In Fig. 4, there are some untrue MVs in the backscreen area when MVS algorithm is not applied. In fig.4 (b), MVS method in [11] generates incorrect MV during the MVS process; therefore the male dancer area is blurred. MVS used in [15], [9] and the proposed algorithm do not produce any blurring artifacts as shown in Fig 4 (c), (d), and (e). However MVS in [15] produces more incorrect MVs and it causes blocking artifact. From the objective and subjective results, the proposed MVS method shows the best performance than the other MVS algorithms.

IV. CONCLUSION

In this paper, we proposed advance motion vector smoothing method. The proposed algorithm detect the outliers using the distance of MVs in the 3x3 size of motion vector window, and corrects the outlier MVs by combining inlier neighboring motion vectors with adaptive weights. From the experimental results, the proposed method gives the best performance in terms of PSNR and subjective quality. The proposed MVS algorithm can be adopted other FRUC algorithms and it can help achieve further performance improvement.

REFERENCES

[1] C. Cafforio, F. Rocca, and S. Tubaro, "Motion compensated image interpolation," *Communications, IEEE Transactions on,* vol. 38, pp. 215-222, 1990.

[2] K. Hilman, H. W. Park, and Y. Kim, "Using motion-compensated frame-rate conversion for the correction of 3: 2 pulldown artifacts in video sequences," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 10, pp. 869-877, 2000.

[3] S.-J. Kang, D.-G. Yoo, S.-K. Lee, and Y. H. Kim, "Multiframe-based bilateral motion estimation with emphasis on stationary caption processing for frame rate up-conversion," *Consumer Electronics, IEEE Transactions on,* vol. 54, pp. 1830-1838, 2008.

[4] B.-D. Choi, J.-W. Han, C.-S. Kim, and S.-J. Ko, "Frame rate up-conversion using perspective transform," *Consumer Electronics, IEEE Transactions on,* vol. 52, pp. 975-982, 2006.

[5] G. De Haan, P. W. Biezen, H. Huijgen, and O. Ojo, "True-motion estimation with 3-D recursive search block matching," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 3, pp. 368-379, 388, 1993.

[6] K.-M. Yang, M.-T. Sun, and L. Wu, "A family of VLSI designs for the motion compensation block-matching algorithm," *Circuits and Systems, IEEE Transactions on,* vol. 36, pp. 1317-1325, 1989.

[7] R. Castagno, P. Haavisto, and G. Ramponi, "A method for motion adaptive frame rate up-conversion," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 6, pp. 436-446, 1996.

[8] V. E. Seferidis and M. Ghanbari, "General approach to block-matching motion estimation," *Optical Engineering,* vol. 32, pp. 1464-1474, 1993.

[9] S.-J. Kang, K.-R. Cho, and Y. H. Kim, "Motion compensated frame rate up-conversion using extended bilateral motion estimation," *Consumer Electronics, IEEE Transactions on,* vol. 53, pp. 1759-1767, 2007.

[10] B.-T. Choi, S.-H. Lee, and S.-J. Ko, "New frame rate up-conversion using bi-directional motion estimation," *Consumer Electronics, IEEE Transactions on,* vol. 46, pp. 603-609, 2000.

[11] T. Ha, S. Lee, and J. Kim, "Motion compensated frame interpolation by new block-based motion estimation algorithm," *Consumer Electronics, IEEE Transactions on,* vol. 50, pp. 752-759, 2004.

[12] J. Zhai, K. Yu, J. Li, and S. Li, "A low complexity motion compensated frame interpolation method," in *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, 2005, pp. 4927-4930.

[13] D. Wang, L. Zhang, and A. Vincent, "Motion-compensated frame rate up-conversion—Part I: Fast multi-frame motion estimation," *Broadcasting, IEEE Transactions on,* vol. 56, pp. 133-141, 2010.

[14] D. Kim and H. W. Park, "An Efficient Motion-Compensated Frame Interpolation Method Using Temporal Information for High-Resolution Videos."

[15] D.-G. Yoo, S.-J. Kang, and Y. H. Kim, "Direction-select motion estimation for motion-compensated frame rate up-conversion," *Display Technology, Journal of,* vol. 9, pp. 840-850, 2013.

[16] S.-J. Kang, S. Yoo, and Y. H. Kim, "Dual motion estimation for frame rate up-conversion," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 20, pp. 1909-1914, 2010.

[17] G. Dane and T. Q. Nguyen, "Motion vector processing for frame rate up conversion," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, 2004, pp. iii-309-12 vol. 3.