An efficient frame rate up-conversion method with adaptive motion estimation and compensation for mobile projection displays

Jong Ok Lee, Seul Ki Jang, Qiao Song Chen, Choon-Woo Kim
School of Information and Communication Eng., Inha University, Incheon, 402-751 Korea
TEL: 82-32-860-7401, e-mail: cwkim@inha.ac.kr

Keywords: frame rate, mobile display, projection, motion estimation

Abstract

Recently, mobile video communication is getting more and more popular. Visual quality and computational complexity are primary factors affecting performance of video communication. Frame rate up-conversion (FRC) is necessary for achieving high visual quality in mobile projection displays. In this paper, a FRC method using motion compensation based on block matching algorithm (BMA) with adaptive block size is proposed. In order to improve the accuracy of the estimated motion vectors, the motion vector refinement technique is proposed. Experiment results indicate that the proposed technique exhibits better performance with lower hardware complexity compared to the conventional methods.

1. Introduction

Nowadays, the terrestrial digital multimedia broadcasting (T-DMB) capable mobile phone is becoming popular. However, due to the reduced frame rate, visual quality of T-DMB on mobile phone needs improvements. Specially, motion judder can be noticed on moving subtitles. In order to improve visual quality of T-DMB on the mobile phone displays, FRC algorithm is employed in this paper. Computational complexity and size of memory resources is quite important in mobile applications.

FRC algorithms have been widely studied for reducing the motion blur on LCDs and/or converting the video format[1-6]. Simple FRC algorithms such as frame repetition and averaging would cause the motion jerkiness and blurring of moving objects. To alleviate these artifacts, the motion compensated FRC (MC-FRC) algorithms have been developed. In order to reduce the computational complexity of the motion estimation (ME), many fast BMA methods have been developed. They include 2D-logarithm search (LOGS)[1], three-step search (TSS)[2], and sub-sampled full search (SSFS)[3], etc.

However, these fast algorithms lack the accuracy of motion vectors. To overcome these problems, various ME methods are proposed. Multi-size blocks are used according to similarity among motion vectors of neighboring blocks in [4]. The MC-FRC using weighted motion error is proposed in [5]. The overlapped block motion compensations are employed in [6]. However, methods in [4-6] demand the heavy computational burden and consequently huge hardware cost. Therefore, it is difficult to be implemented in real-time applications.

In this paper, we propose a FRC method to improve the visual quality for mobile projection displays. In order to estimate motion vectors, adaptive block size and search window depending on the motion activity level are proposed. To improve the accuracy of motion vectors, the motion vector refinement technique is proposed. The proposed method achieves better visual quality performance with low complexity to be utilized for mobile projection displays.

In section 2, the proposed method is explained in details. In section 3, the experimental results are presents. Finally, conclusion is addressed in section 4.

2. Proposed frame rate conversion

Fig. 1 shows the processing flow when the proposed method is utilized for T-DMB receiving mobile phone. The T-DMB source, size of QVGA with frame rates of 30 fps, is transmitted to T-DMB receiver. The proposed method is applied to double the frame rates of T-DMB.

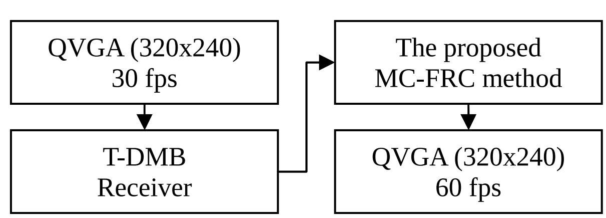

Fig. 1. Processing flow of the proposed method

The flow chart of the proposed method is shown in Fig. 2. In case of frame transition such as scene change, the interpolated frame is obtained without estimating the motion vectors. When decided as non-transition type, the proposed MC-FRC is performed. It consists of three steps. First, the motion of objects in a scene is estimated based on the BMA. Second, the motion vector refinement is performed to improve the accuracy of motion vectors. Third, the motion compensated interpolation is applied.

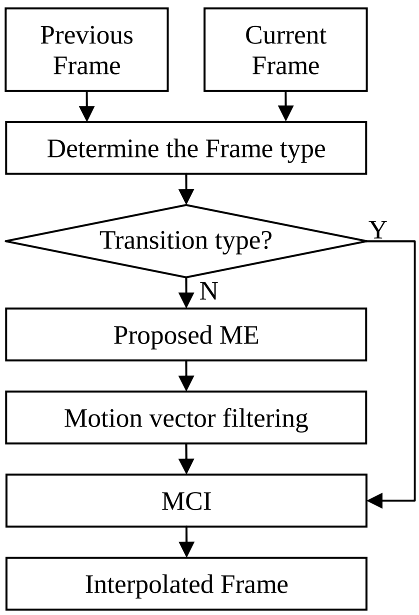

Fig. 2. Flow chart of the proposed MC-FRC method

2.1. Proposed motion estimation

Fig. 3 shows the flow chart of the proposed ME method. In the proposed method, the size of block and search window are determined according to the motion activity level for each of sixteen 8x8 blocks. The motion activity level for each block is estimated by the summed absolute difference (SAD) between the current and previous frames. The motion activity level L for an NxM block is estimated by

$$L = \sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\left|F_n(i, j) - F_{n-1}(i, j)\right| \qquad (1)$$

where $F_n(i, j)$ is the gray level at location (i, j) in the nth frame. If L is larger than the threshold, then the block is determined as motion type. Otherwise, it is determined as non-motion type.

Block size for estimating motion vector is determined based on the block types of sixteen 8x8 blocks. When neighboring blocks are no-motion type, the 8x8 blocks is merged into 8x16 or 16x8 or 16x16 or 32x32 block. Size of search window is determined by the block type. If the processing block is no-motion type, then the search window is ± 2, otherwise search window is ± 7.

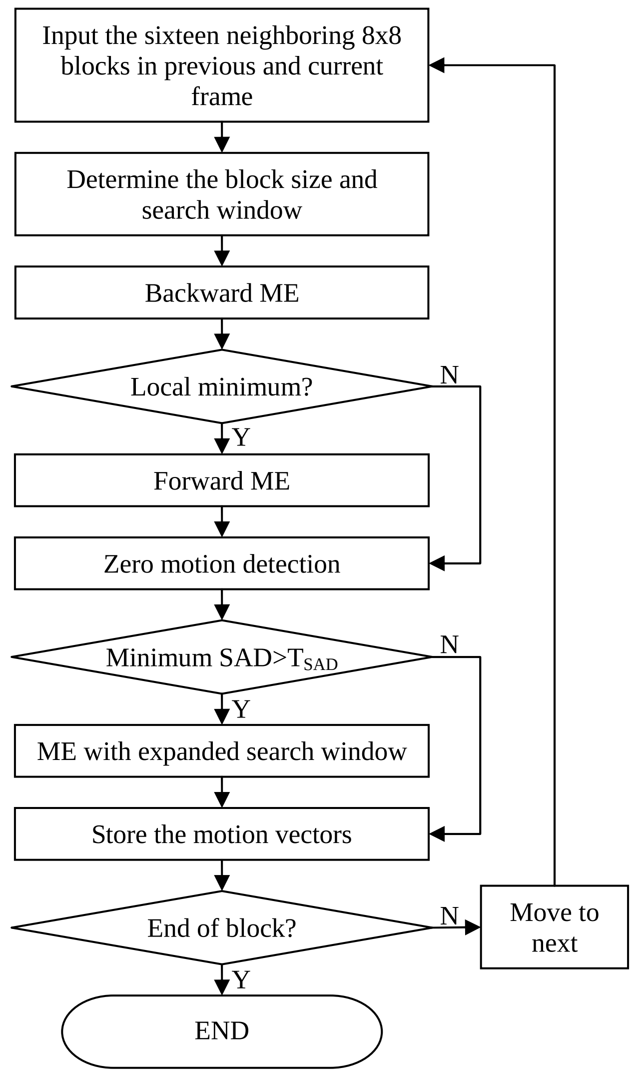

Fig. 3. Flow chart of the proposed ME method

Next step is to perform the backward ME based on the determined block size and search window. In the backward ME, current frame is used as reference. Error criterion for matching is SAD given as

$$SAD(x, y) = \sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\left|F_n(i, j) - F_{n-1}(i + x, j + y)\right| \qquad (2)$$

where (x, y) represents displacement from reference block. The motion vector (v_i, v_j) of a given block is obtained by

$$(v_i, v_j) = \arg \min_{(x,y)\in S} \{SAD(x,y)\} \qquad (3)$$

where S denotes the search window. In order to avoid being trapped at a local minimum, forward ME is performed. In the forward ME, the previous frame is used as the reference.

In the case of no motion occurred between two consecutive frames, estimated motion vector may not be $(0,0)$. 'Zero motion estimation' is performed to solve this problem. If the difference of $SAD(0, 0)$ and $SAD(v_i, v_j)$ is less than threshold value, the motion vector is set to zero. After 'zero motion estimation' is performed, it is examined whether the motion of block is greater than the size of search window by comparing $SAD(v_i, v_j)$ with the predetermined threshold T_{SAD}. For the motion greater than the search window, the ME is performed on the expanded search window.

2.2 Proposed motion vector refinement method

Fig. 4 shows the flow chart of the proposed motion vector refinement method. In this paper, the similarity is determined according to the difference between the motion vector components in the current and the neighboring blocks. In Fig. 5, the window of the cross shape, marked with bold lines, represents the neighboring blocks. V_0 and V_i describes the current motion vector and the neighboring motion vectors, respectively. Among the four calculated differences, the zero value is searched. If the value of zero is detected, it means that the current motion vector is similar to the neighboring motion vector. Otherwise, median filtering is applied to the estimated motion vectors.

Next step is to determine the reliability of the motion vector in the current block. In this paper, the following criterion, *HSAD*, is utilized to represent the reliability.

$$HSAD = \sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\left| F_n\left(i+\frac{v_i}{2}, j+\frac{v_j}{2}\right) - F_{n-1}\left(i-\frac{v_i}{2}, j-\frac{v_j}{2}\right)\right| \qquad (4)$$

If the calculated reliability is smaller than the predetermined threshold value, it is assumed that the estimated motion vector is reliable. Otherwise, it is regarded as unreliable motion vector. When determined as unreliable, *HSAD* defined in Eq. (4) is calculated for each of four neighboring blocks shown in Fig. 5. Current motion vector is replaced by that having the minimum *HSAD* value.

This procedure is repeated until all the motion vectors are determined.

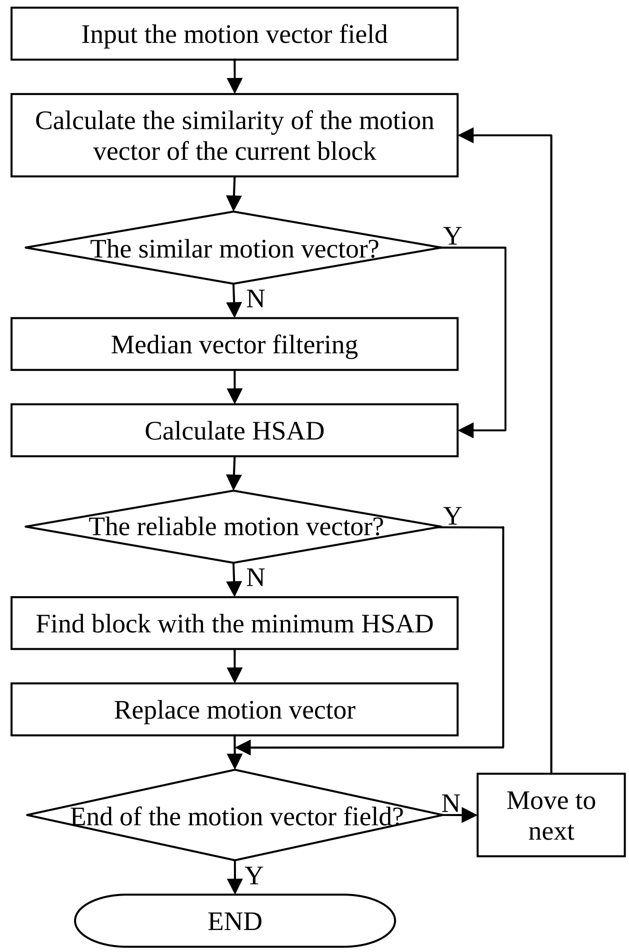

Fig. 4. Flow chart of the proposed motion vector refinement method

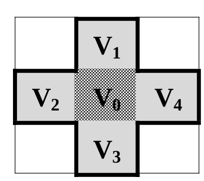

Fig. 5. Four neighboring block

Once the motion vectors are refined, the MCI is performed. The interpolated frame is calculated by the following formula.

$$F_{n-\frac{1}{2}}(i,j) = \frac{F_n\left(i+\frac{v_i}{2}, j+\frac{v_j}{2}\right) + F_{n-1}\left(i-\frac{v_i}{2}, j-\frac{v_j}{2}\right)}{2} \qquad (5)$$

where (v_i, v_j) indicates the refined motion vector.

3. Experimental Results

In order to evaluate the performance of the proposed method, experiments are conducted for the three video sequences: 'Hall monitor', 'Foreman', and 'Football'. 'Hall monitor' consists of low spatial frequency contents and represents slow motion. 'Football' has relatively fast motion with high spatial details. 'Foreman' represents speed of motion and frequency contents ranging in the middle of 'Hall monitor' and 'Football. Each of three sequences consists of 100 frames. The even numbered frames are used as input for FRC. The odd numbered frames are interpolated based on the even numbered frames using the proposed method. Peak signal-to-noise ratio (PSNR) is used for comparing the proposed MC-FRC method with the TSS [2], full search (FS) with fixed block size (16x16), and MC-FRC with multi-size blocks [4]. The search window is set to be ± 7. Table 1 shows the average PSNR of the interpolated frames. Computational complexity of the proposed method is much simpler than that of the method in [4]. As shown in Table 1, the proposed ME method has similar performance compared to the method in [4]. However, the proposed ME with motion vector refinement method shows the best results for all of three image sequences used in the experiments.

Figure 6 shows an example of the wired expressions of the block size of the interpolated results. In the still regions, large blocks are used to estimate the motion vector. In regions with motions, relatively smaller block size is chosen.

4. Conclusion

In this paper, a frame rate up-conversion method is proposed. The objective of the proposed method is to improve visual quality performance with low computational complexity. The proposed method can be utilized for mobile projection displays. Adaptive block size and search window are utilized. After the ME process, the estimated motion vector is refined. Based on these motion vectors, motion compensated interpolation is performed. Experiment results indicate that the proposed technique exhibits better performance with lower hardware complexity compared to the conventional methods.

Table 1. Comparisons of average PSNR [dB]

	Hall monitor	Foreman	Football
FS	32.21	31.54	20.34
TSS	31.59	28.06	19.59
Method in [4]	32.69	32.41	21.04
Proposed ME	32.54	32.37	20.88
Proposed ME + motion vector refinement	33.47	32.81	21.24

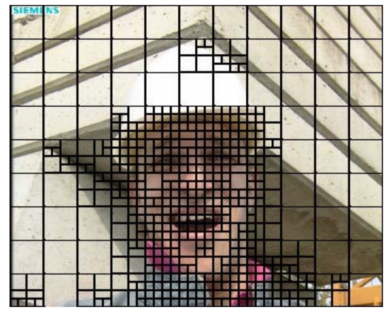

Figure 6. Expression of used block size

5. Reference

[1] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding", IEEE Trans. Comm., Vol. COM-29, pp. 1799-1806 (1981)

[2] T. Koga et. al., "Motion-compensated interframe coding for video conferencing", Proc. NTC 81, pp. C9.6.1-9.6.5 (1981)

[3] B. Liu and A. Zaccarin, "New fast algorithms for the estimation of block motion vectors", IEEE Trans. Circuits Sys. & Video Technol., Vol. 3, pp. 148-157 (1993)

[4] S. Fujiwara and A. Taguchi, "Motion-compensated frame rate up-conversion based on block matching algorithm with multi-size blocks", ISPACS 05, pp. 353-356 (2005)

[5] S. H. Lee, O. Kwon, and R. H. Park, "Weighted-adaptive motion-compensated frame rate up-conversion," IEEE Trans. Consumer Electronics, vol. 49, pp. 485-492 (2003)

[6] O. A. Ojo and G. de Haan, "Robust motion-compensated video upconversion", IEEE Trans. Consumer Electronics, Vol. 43, pp. 1045-1045 (1997)

Acknowledgement

This work was supported by the IT R&D program of MIC/IITA. [2006-S053-01, Embedded LPD Module for HHP]