Triple Motion Estimation and Frame Interpolation based on Adaptive Threshold for Frame Rate Up-Conversion

Hanieh Naderi and Mohammad Rahmati, Member, IEEE

Abstract—In this paper, we propose a novel motion-compensated frame rate up-conversion (MC-FRUC) algorithm. The proposed algorithm creates interpolated frames by first estimating motion vectors using unilateral (jointing forward and backward) and bilateral motion estimation. Then motion vectors are combined based on adaptive threshold, in order to create high-quality interpolated frames and reduce block artifacts. Since motion-compensated frame interpolation along unilateral motion trajectories yields holes, a new algorithm is introduced to resolve this problem. The experimental results show that the quality of the interpolated frames using the proposed algorithm is much higher than the existing algorithms.

Index Terms—Frame rate up-conversion, frame interpolation, motion estimation, motion compensation.

I. INTRODUCTION

In computer video displays, television, and movie data, the frame rate is the number of frames that are displayed per second. The higher the number of frames displaying per second, the smoother the video playback appears to the user [1]. In order to improve the temporal quality and jerkiness artifacts in video and the hold-type display, such as liquid crystal display (LCD), frame rate up-conversion (FRUC) interpolates new frames between original frames and increases the number of frames [2]. FRUC is also used in video compression for reconstructing the skipped frames [3].

FRUC algorithms are generally classified into two methods. The first method generates the interpolated frames using frame repetition and frame averaging from the pixels in the previous and the next frame. The methods in these categories are used because of their simplicity and low complexity. However, these simple methods produce insufficient results that lead motion blurring and ghost artifacts in the interpolated frame [4], [5]. The second method is introduced to solve this problem and is called motion-compensated frame rate up-conversion (MC-FRUC) [6].

MC-FRUC consists of two steps: motion estimation (ME) and motion-compensated interpolation (MCI). Motion estimation tracks motion trajectories between the previous and the next frame and estimates motion vectors. Motion-compensated interpolation uses these motion vectors to produce interpolated frames. The quality of interpolate frames clearly depend on motion estimation and motion-compensated interpolation.

Several motion estimation algorithms such as block matching algorithm (BMA) [7], three step search [8], 2D logarithmic search [9] and diamond search [10] are proposed. Among the noted algorithms, block matching algorithm is the most popular algorithms. Although it is more complex than other fast algorithms, but it can find the most accurate motion in compare to others. Moreover, architecture of [11] can implement the high-speed processing of block matching algorithm by parallel computation.

The traditional block matching algorithm (also known as unilateral motion estimation [12]-[19]), divides one of two neighboring frames into non-overlapped blocks and estimates motion vectors of each block by calculating the sum of absolute difference (SAD) between non-overlapped block in the first frame and each possible block in the search range in the other frame and finally generates interpolated frame along the motion vectors. Unilateral method can creates holes and overlaps in some pixels in the interpolated frame, due to no motion vector and multiple motion vector, respectively. Hence, different algorithms are proposed to handle these problems. For example, to handle overlaps, simple algorithms such as averaging and overwriting on the overlapped pixels are used in [6], [12]. To fill holes, a median filter is used in [18]. In [6], [12] holes are covered by the pixel values from the previous or the next frame.

To solve the problems of overlaps and holes, bilateral motion estimation method is proposed. Bilateral motion estimation [20]-[25] divides the frame to be interpolated into non-overlapped blocks before it is really created and then estimates a motion vector using the temporal symmetry between blocks of the previous and the next frames in the search range around initial block position, which pass through block in the interpolated frame. Since bilateral motion estimation assign one motion vector to each non-overlapped block, overlaps and holes are not generated. However, there is essential problem in existing bilateral methods, when the sum of absolute difference between each two blocks of previous and next frames within the search range is obtained. Motion trajectory of two blocks which
have the minimum SAD is taken as the true motion vector. Some of motion vectors may be inaccurate in the uniform or periodic regions. So, if the obtained motion vectors are directly employed for motion compensated interpolation, visual artifacts are observed in the interpolated frames. Hence, to obtain more accurate motion vectors, motion vector smoothing algorithms are proposed [26]-[28], to correct the false motion vectors to enhance the interpolated frame quality.

In this paper, we propose a novel MC-FRUC algorithm that employs unilateral and bilateral methods simultaneously, to benefit advantages of both methods. The proposed algorithm, first performs bilateral motion estimation and then using vector median filter to obtain more accurate motion vectors. Then overlapped block motion compensation (OBMC) [29], [30] is employed to reduce block artifacts. Second, with combination of forward and backward motion estimation, reduces the number of holes in unilateral motion estimation. To remove the residual holes, the proposed algorithm uses the interpolated frame which obtained from bilateral method. Finally, two interpolated frames which obtained from bilateral and unilateral methods are combined based on adaptive threshold to obtain a higher quality interpolated frame. Fig. 1 shows the block diagram of the proposed MC-FRUC algorithm.

Fig. 1. Block diagram of the proposed MC-FRUC algorithm.

The rest of this paper is organized as follows. Section II describes the proposed algorithm and Section III, considers experimental results. Finally, Section IV concludes this paper.

II. PROPOSED MC-FRUC ALGORITHM

The proposed algorithm uses the unilateral and bilateral methods simultaneously, to benefit advantages of both methods. Using bilateral method, we solve holes and overlaps problem, but temporal symmetry problem remains. Then, vector median filter (VMF) [31] and overlapped block motion compensation (OBMC) [30] is used to solve the recent problem.

When the unilateral method fails to yield acceptable results, due to holes and overlap areas, the proposed algorithm uses a combinations of forward and backward motion estimation to reduce the number of holes. Finally, a bilateral method is used to remove the remaining holes.

In the proposed algorithm, two interpolated frames which obtained from bilateral and unilateral methods are combined based on adaptive threshold to improve the quality of interpolate frame. The details are given in the following subsections.

A. Motion Estimation

The bilateral and unilateral motion estimation methods are simultaneously used to estimate the initial motion vector fields.

In bilateral motion estimation, firstly we divide the frame to be interpolated into non-overlapped blocks before it is really created. Then we select a linear trajectory between the previous and the next frames passing at the center of the each block in the interpolated frame as shown in Fig. 2. We use sum of absolute difference (SAD) as the motion vector validity measure that is defined as,

\[
SAD = \sum_{i=1}^{m} \sum_{j=1}^{n} |f_p(i, j) - f_n(i, j)|
\]  

(1)

where \(m\) and \(n\) denote the horizontal length and vertical length of candidate blocks, respectively. The \(f_p\) denotes the pixel value of the candidate block in the previous frame and \(f_n\)
denotes the pixel value of the candidate block in the next frame and \((i,j)\) represents the pixel location in the candidate block. The search range is limited to a small displacement around the initial block position. Motion vectors are found in accordance with the smallest SAD of two candidate blocks of previous and next frames in the search range. The motion vectors for each block in the interpolated frame is obtained. These motion vectors between the interpolated frame and previous and next frames are symmetric. Bilateral motion vector fields can be obtained by this motion estimation method.

**B. Motion Vector Smoothing**

As noted in the previous section, bilateral motion vector fields are obtained by bilateral motion estimation. The motion vector field cannot exactly indicate motion of objects, since the property of the temporal symmetry of bilateral motion estimation leads to estimate of false motion vectors. The false motion vectors cause extremely degraded frame quality. We smooth motion vector field and remove outliers in the motion vector field, using vector median filtering. In fact, each candidate motion vector is replaced by a median of candidate motion vector and its eight neighboring motion vectors.

**C. Motion Compensated Interpolation**

Once the final motion vector field is obtained, the interpolated frame can be obtained using bilateral motion-compensated interpolation algorithm as follows,

\[
f_{\text{bi}}(i, j) = \frac{1}{2} \left( f_p(i + m_{vx}, j + m_{vy}) + f_p(i - m_{vx}, j - m_{vy}) \right),
\]

where \(f_{\text{bi}}\) denotes the pixel value of the bilateral interpolated frame, and \(mv = (m_{vx}, m_{vy})\) denotes motion vector after motion vector smoothing.

After constructing the interpolated frame, overlapped block motion compensation (OBMC) is finally applied to reduce block artifacts in the interpolated frame. As shown in Fig. 4, each block in the interpolated frame is considered an enlarged block which overlap with neighboring blocks.

**Fig. 3.** Unilateral motion estimation (forward motion estimation and backward motion estimation.) (a) Forward motion estimation. (b) Backward motion estimation.

**Fig. 4.** Overlapped block motion compensation.

In unilateral motion estimation we use forward motion estimation and backward motion estimation. In forward motion estimation, firstly we divide previous frame into non-overlapped blocks and then define a search range around the initial block. Each block in the previous frame is compared with all of the blocks in the search range in the next frame. The block which have the least SAD with block in the previous frame is chosen and its motion vector is selected. The backward motion estimation derived in the same way. As shown in Fig. 3, overlaps and holes are created according motion vector trajectory. The gray regions in the interpolated frame show the holes and the overlapped area between the small squares in the interpolated frame are called overlaps.

Forward and backward motion vector fields can be obtained by forward and backward motion estimation methods.

Consider region 1 in Fig. 4 which overlaps four blocks \((B_1, B_2, B_3, B_4)\). To obtain a pixel \(f_{\text{bi}}(i, j)\) in region 1, the following is used,

\[
f_{\text{bi}}(i, j) = \frac{1}{8} \left( \sum_{x=0}^{3} \sum_{y=0}^{3} f_p(i + 2x m_{vx}, j + 2y m_{vy}) \right),
\]

where \(mv = (m_{vx}, m_{vy})\) denotes for motion vectors of four neighboring blocks \((B_1, B_2, B_3, B_4)\) in region 1.

Any pixel in region 2, which overlaps two blocks \((B_1, B_2)\), is obtained by,
\[
\begin{align*}
\text{BlockSAD AdaptiveThreshold} \\
&= f_i, j (f_i, j - f_i, j (i, j)) \\
&= \begin{cases} 
1 & \text{if } f_i, j = H \text{ and } f_i, j \neq H \\
1 & \text{if } f_i, j = H \text{ and } f_i, j = H \\
1 & \text{if } f_i, j \neq H \text{ and } f_i, j \neq H \\
2 & \text{otherwise} 
\end{cases}
\end{align*}
\]

where \( H \) is a constant that indicates the pixels are within holes and \((i,j)\) represents the pixel location. If one pixel has only one existing value, because there is a hole in either forward interpolated frame, \( f_i \), or backward interpolated frame, \( f_b \), this existing value is taken as the ultimate value in the final interpolated frame, \( f_o \). If one pixel has two existing values from \( f_i \) and \( f_b \), the proposed algorithm applies the average of the two existing values as the ultimate value in the final interpolated frame. If there is no existing value from \( f_i \) and \( f_b \), the proposed algorithm applies existing pixel in the same position in the bilateral interpolated frame, \( f_{bo} \), as the ultimate value in the final interpolated frame, as shown in Fig. 5(c) and (f).

Finally, the proposed algorithm makes the final interpolated frame, \( f_o \), defined as,

\[
\begin{align*}
\text{BlockSAD} \geq \text{AdaptiveThreshold} \\
f_i, j = \begin{cases} 
1 & \text{if } f_i, j (i, j) \\
2 & \text{else} 
\end{cases}
\end{align*}
\]

where \( \text{BlockSAD} \) represents the SAD of two candidate blocks of the previous and the next frames which their averages are replaced with the desired interpolated block, and the \textit{adaptive threshold} is defined as the average of the \textit{BlockSAD}s of blocks which are located before the desired interpolated block.

If \( \text{BlockSAD} \) of desired interpolated block is considered to be large, the desired interpolated block is assumed unreliable and quality of final interpolated frame is degraded. We use \textit{adaptive threshold} as a comparison criterion, to improve the results. If the \( \text{BlockSAD} \) is larger than \textit{adaptive threshold}, we set the desired interpolated block to average of the bilateral, forward, and backward interpolated frames. Otherwise, we set the desired interpolated block to weighted average of these frames. In this weighted average, we assign more weight to bilateral interpolated frame.
III. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed algorithm, we conducted experiments using twelve standard Common Intermediate Format (CIF) test sequences. All of the test sequences have a frame rate of 30 frames per second. We considered the first 102 frames of each test sequence. To obtain quality of the interpolated frames, we removed 50 odd frames (from frame 3 till frame 101) and interpolated them using 51 even frames. Then the peak signal-to-noise ratio (PSNR) values were computed from the difference between the original odd frames and the interpolated frames. Considerably, Stefan sequence just included 90 frames and so its PSNR is based on 44 frames. For experiments, in unilateral motion estimation, we set the block size to 8×8 pixels and the search range to ±16 pixels. Temporal symmetry is one of the characteristics of bilateral motion estimation, so large search range always leads to artifacts in the interpolated frame where some blocks in the scene are replaced with the background. Hence, conventional algorithms which use bilateral motion estimation, need to limit the search range with a smaller range. For instance, in [21], [24], the block size is 16×16 and the search range is ±8. Therefore, we set the block size to 16×16 pixels and the search range to ±8 pixels for bilateral motion estimation. For OBMC, each block is enlarged from 16×16 to 20×20 to reduce block artifacts.

A. Subjective evaluation

We compared interpolated results of the proposed algorithm with bilateral method and unilateral method (jointing of forward and backward methods). Fig. 6 and Fig. 7 illustrate interpolated frames of the test sequences, Stefan and Tennis. We created weighted combination of unilateral and bilateral methods, based on adaptive threshold which yields a better results than each of unilateral and bilateral method alone. If bilateral method had better results in some blocks, its weight is higher than the weight of unilateral method.

B. Objective evaluation

Table I illustrates the average PSNRs of 50 frames that interpolated by unilateral method, bilateral method and proposed algorithm. As seen in the Table, the proposed algorithm using combination of unilateral and bilateral methods yield a 1dB and 0.75dB average PSNR improvement compared with unilateral and bilateral method, respectively.

In order to evaluate the performance of the proposed algorithm, we compared it to the conventional unilateral FRUC [17] and conventional bilateral FRUC [21] which the test conditions were the same as in our experiments.

| Test sequences | unilateral | bilateral | Proposed algorithm |
|----------------|------------|-----------|---------------------|
| Foreman        | 33.30      | 33.47     | 34.17               |
| Football       | 21.92      | 22.00     | 22.39               |
| Mobile         | 24.94      | 28.82     | 26.79               |
| Flower         | 29.52      | 29.79     | 30.40               |
| Stefan         | 27.14      | 27.46     | 28.03               |
| Coastguard     | 30.02      | 32.16     | 32.02               |
| Paris          | 32.32      | 32.16     | 33.13               |
| Soccer         | 28.71      | 29.14     | 29.72               |
| Tennis         | 28.78      | 28.67     | 29.32               |
| Akiyo          | 44.00      | 45.14     | 45.02               |
| News           | 35.36      | 36.15     | 36.38               |
| Silent         | 36.00      | 36.04     | 36.64               |
| Average        | 31.00      | 31.75     | 32.00               |
Table II presents the average PSNRs of the three algorithms. As seen in Table II that our algorithm achieves an improvement of more than 0.7dB and 1.5dB than the conventional unilateral and bilateral FRUC, respectively.

Fig. 8 indicates the PSNRs for the interpolated frames in four test sequences: Foreman, Flower, Silent and Stefan. The graphs indicate the proposed algorithm are effective for entire frames.

| Test sequences | Conventional unilateral FRUC [17] | Conventional bilateral FRUC [21] | Proposed algorithm |
|----------------|-----------------------------------|----------------------------------|--------------------|
| Foreman        | 32.71                             | 31.82                            | 34.17              |
| Football       | 22.17                             | 21.31                            | 22.39              |
| Mobile         | 26.18                             | 25.39                            | 26.79              |
| Flower         | 27.93                             |                                 | 30.40              |
| Coastguard     | 30.75                             |                                 | 32.02              |
| Akiyo          | 45.11                             |                                 | 45.02              |
| News           | 37.25                             |                                 | 36.38              |
| Silent         | 35.81                             |                                 | 36.64              |
| Average        | 32.24                             | 26.17                            | 32.98              |

IV. CONCLUSION

In this paper, we proposed a novel MC-FRUC algorithm. The proposed algorithm combined unilateral and bilateral methods based on adaptive threshold to build a high quality of interpolated frames for MC-FRUC. Since motion-compensated frame interpolation along unilateral motion trajectories yields holes, this paper presented a new algorithm to fill holes. Experimental results showed that the proposed algorithm performed superior than existing algorithms both in terms of subjective and objective criterion evaluations.

REFERENCES

[1] Y. Dar and A. M. Bruckstein, "Motion-Compensated Coding and Frame Rate Up-Conversion: Models and Analysis", IEEE Trans. Image Processing, vol. 24, no. 7, pp. 2051-2066, Jul. 2015.
[2] M. Klopmenhower and L. J. Velthoven, “Motion blur reduction for liquid crystal displays: Motion compensated inverse filtering,” in Proc. SPIE-IS&T Electron. Imag. 2004, vol. 5308, pp. 690–699, Jan. 2004.
[3] B. Catarina, J. Ascenso and F Pereira. "Side information creation for efficient Wyner-Ziv video coding: classifying and reviewing." Signal Processing: Image Communication vol. 28, No. 7: pp. 689-726, May, 2013.
[4] J. M. Boyce, “Noise reduction of image sequences using adaptive motion compensated frame averaging” In icassp, IEEE, pp. 461–464, March. 1992.
[5] T. Y. Kuo, J. Kim and C. J. Kuo, "Motion-compensated frame interpolation scheme for H. 263 codec" in Proc. Int. Symposium. Circuits and Systems, vol. 4, pp. 491–494, 1999.
[6] C. L. Huang and T. T. Chai, “Motion-compensated interpolation for scan rate up-conversion” Opt. Eng., vol. 35, no. 1, pp. 166–176, Jan. 1996.
F. Dufaux and F. Moscheni, “Motion estimation techniques for digital TV: A review and a new contribution,” Proc. IEEE, vol. 83, pp. 858–876, June 1995.

Jing, Xuan, and Lap-Pui Chau. "An efficient three-step search algorithm for block motion estimation." IEEE Trans. Multimedia, vol. 6, pp. 435–438, June 2004.

Jain, J., Jain, A. “Displacement measurement and its application in interferometric image coding” IEEE Trans. Commun., vol. 29, pp. 1799–1808, Dec. 1981.

Zhu, S., Ma, K.-K. “A new diamond search algorithm for fast block matching motion estimation” IEEE Trans. Image Process. vol. 9, pp. 287–290, Feb. 2000.

L. Y. Kang and L. G. Chen. “A data-interlacing architecture with two-dimensional data-reuse for full-search block-matching algorithm.” IEEE Trans. Circuits Syst. Video Technol. vol. 8.2, pp. 124-127, Apr. 1998.

S.-C. Han and J. W. Woods, “Frame-rate up-conversion using transmitted motion and segmentation fields for very low bit-rate video coding,” in Proc. Int. conf. Image Processing, Washington, DC, vol. 1, pp. 747–750, Oct. 1997.

S. Liu, C.-C. J. Kuo, and J. W. Kim, “Hybrid global-local motion compensated frame interpolation for low bit rate video coding,” Journal of Visual Communication and Image Representation, vol. 14, pp. 61–79, Feb. 2003.

T. Liu, K.-T. Lo, J. Feng, and X. Zhang, “Frame interpolation scheme using inertia motion prediction.” Signal Processing: Image Communication vol. 18, pp. 221–229, Jan. 2003.

L. Hongbin, X. Ruiqin, Z. Debin, S. Ma and G. Wen “Multiple hypotheses Bayesian frame rate up-conversion by adaptive fusion of motion-compensated interpolations” IEEE Trans. Circuits Syst. Video Technol. vol. 22, pp. 1188–1198, Aug. 2012.

W. Demin, A. Vincent, P. Blanchfield, and R. Klepko. "Motion-compensated frame rate up-conversion—Part II: New algorithms for frame interpolation." IEEE Trans. Broadcast. vol. 56, no. 2, pp. 142-149, Jun. 2010.

K. U. Scob, and M. H. Sunwoo. "New frame rate up-conversion algorithms with low computational complexity." IEEE Trans. Circuits Syst. Video Technol. vol. 24, no. 3 pp. 384-393, March. 2014.

B. P. Jenny and H. Nicolas. "A new method for region-based depth ordering in a video sequence: application to frame interpolation." Journal of Visual Communication and Image Representation. vol. 13, no. 3, pp. 363-385, Aug. 2002.

T. Wang, C. L. Zhang, Y. He, and Y. P. Tan. "Frame rate up-conversion using trilateral filtering." IEEE Trans. Circuits Syst. Video Technol. vol. 20, no. 6, pp. 886-893, Jun. 2010.

T. Ha, S. Lee, and J. Kim, “Motion compensated frame interpolation by new block-based motion estimation algorithm,” IEEE Trans. Consum. Electron. vol. 50, no. 2, pp. 752–759, May 2004.

S. J. Kang, K. R. Cho, and Y. H. Kim, “Motion compensated frame rate up-conversion using extended bilateral motion estimation,” IEEE Trans. Consum. Electron., vol. 53, no. 4, pp. 1759–1767, Nov. 2007.

B. D. Choi, J. W. Han, Ch. S. Kim, and S. J. Ko. "Motion-compensated frame interpolation using bilateral motion estimation and adaptive overlapped block motion compensation." IEEE Trans. Circuits Syst. Video Technol. vol. 17, no. 4, pp. 407-416, April. 2007.

S. J. Kang, S. J. Yoo, and Y.-H. Kim "Dual motion estimation for frame rate up-conversion," IEEE Trans. Circuits Syst. Video Technol., vol. 20, no. 12, pp. 1909–1914, Dec. 2010.

Ch. Junhua, and Y. Liu. “Frame rate up-conversion based on dual criterion motion estimation and motion vector adjustment.” Signal, Image and Video Processing. pp. 1-9, July. 2014.

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,” IEEE Trans. Consum. Electron., vol. 54, no. 4, pp. 1830–1838, Nov. 2008.

J. Guo and J. H. Kim. "Adaptive motion vector smoothing for improving side information in distributed video coding.” Journal of Information Processing Systems. vol. 7, no. 1, pp. 103-110, 2011.

B. D. Choi, J. W. Han, C. S. Kim, and S. J. Ko, "Frame rate up-conversion using perspective transform," IEEE Trans. Consumer Electronics, vol. 52, no. 3, pp. 975-982, Aug. 2006.

T. Ha, S. Lee, and J. Kim, “Motion compensated frame interpolation by new block-based motion estimation algorithm,” IEEE Trans. Consumer Electronics, vol. 50, no. 2, pp. 752–759, May. 2004.