NEW FRAME RATE UP-CONVERSION USING BI-DIRECTIONAL MOTION ESTIMATION

Byung-Tae Choi[1], *Student Member, IEEE*, Sung-Hee Lee[2] and Sung-Jea Ko[1], *Senior Member, IEEE*

[1]Department of Electronics Engineering, Korea University, Korea
E-mail: sjko@dali.korea.ac.kr

[2]Digital Media System Lab., Samsung Electronics Co., Korea

Abstract— **We propose a new frame rate up-conversion algorithm for high quality video. In the proposed scheme, bi-directional motion estimation (ME) is performed to construct the motion vector (MV) field for the frame to be interpolated. Unlike conventional motion-compensated interpolation (MCI) algorithms, the proposed technique does not produce any overlapped pixel and hole region in the interpolated frame, and thus can utilize the overlapped block motion compensation technique to reduce the blocking artifacts. The proposed algoritm is very simple to implement on consumer products when compared to conventional MCI methods. Computer simulation shows a high visual performance of the proposed frame rate up-conversion algorithm.**

Keywords— **frame rate up-conversion, true motion, bi-directional motion estimation, overlapped block motion compensated interpolation**

I. Introduction

FRAME rate up-conversion is one of the main issues that have arisen in recent years with the explosive growth of image sources and display formats. For example, currently available motion pictures have a temporal rate of 24, 25, or 30 frames per second, while the HDTV and multimedia PC systems support higher temporal rates to reduce artifacts such as flicker and improve visual image quality [1]. Therefore, frame rate must be up-converted to use motion pictures in the HDTV or multimedia environments. Moreover, frame rate up-conversion technique can be used for video compression and slow motion replay.

Frame rate up-conversion algorithms such as frame repetition and linear interpolation by temporal filtering produce "jerkiness" into the motion portrayal and blurring of object boundaries, respectively [2]. It has been shown that the MCI technique provides the best solution in temporal rate up-sampling applications [3]. For the MCI scheme, the motion vectors must represent the true motion of objects in the image sequence since all the interpolation processes are controlled by the motion vectors. Several algorithms have been proposed for the true-motion estimation [4] - [6]. For example, Thoma and Bierling [4] use hierarchical block matching motion estimation to obtain motion vectors that more closely reflect the true motion of the displaced objects than the full search block-matching algorithm (FS-BMA). Other techniques smooth the motion vector field in spatio-temporal directions to estimate the true object motion [6], [7].

Frame rate up-conversion using block-based motion compensation introduces the overlapped (multi-passing of motion trajectories) and hole (no motion trajectory is passing) regions in the interpolated frame. Kuo *et al.* used median filtering for the overlapped pixels [8]. To handle the holes, the spatial interpolation might be adopted [9]. This method, however, requires complicated operations since the spatial neighborhood of a hole may still contain other holes. Another method to fill the hole is to estimate the motion vector by using the neighboring motion field. A forward/backward prediction method based on the segmentation of the holes into the covered and uncovered regions has been developed under the assumption that the video sequence has static background [8], [10] - [11]. This method, however, produces unsatisfactory results especially in video sequences with camera motion such as panning and zooming which violate the assumption of the static background.

In this paper, a new frame rate up-conversion scheme is proposed to overcome the problem of the overlapped pixels and hole regions. In the proposed scheme, bi-directional motion estimation is performed using the existing previous and current frames to construct the motion vector field for the frame to be interpolated. Unlike conventional motion-compensated interpolation algorithms, the proposed technique does not produce any overlapped pixel and hole region in the interpolated frame, and thus can utilize the overlapped block motion compensation (OBMC) technique to reduce the blocking artifacts.

The paper is organized as follows. In the next section, currently existing motion-compensated frame rate up conversion algorithm and its outstanding problems are described. A new frame rate rate up-conversion scheme is presented in section III. An overlapped block MCI to reduce blocking artifacts is also presented in this section. Experimental results are given and discussed in section IV. Finally, section V colcludes this paper.

II. Review on Conventional MCI's

Let $f_t(\vec{p})$ denote the pixel intensity at 2-D pixel coordinate $\vec{p}$ of the frame at temporal reference t and $\vec{D}_{1,2}(\vec{p})$denote the motion vector at pixel $\vec{p}$ from the previous frame f_{t_1} to the current frame f_{t_2}. Temporal position

Original manuscript received June 19, 2000

ratios W_f and W_b can be defined by

$$W_f = \frac{t_i - t_1}{t_2 - t_1} \text{ and } W_b = \frac{t_2 - t_i}{t_2 - t_1}, \tag{1}$$

where $t_1 < t_2$, for the forward and backward interpolation ratios, respectively.

In the MCI method, a segmentation mask is created between two successive frames. This mask frame is divided into stationary background, moving objects, and covered and uncovered regions as shown in Fig. 1. Then, based on the segmentation information, an appropriate interpolation method is determined and the pixel value of the interpolated frame is predicted as follows:

1. Moving objects: Bi-directional MCI

$$\begin{aligned} f_{t_i}(\vec{p}) &= W_b \cdot f_{t_1}(\vec{p} - W_f \cdot \vec{D}_{1,2}(\vec{p})) + \\ &\quad W_f \cdot f_{t_2}(\vec{p} + W_b \cdot \vec{D}_{1,2}(\vec{p})) \end{aligned} \tag{2}$$

2. Covered background: Forward prediction

$$f_{t_i}(\vec{p}) = f_{t_1}(\vec{p}) \tag{3}$$

3. Uncovered background: Backward prediction

$$f_{t_i}(\vec{p}) = f_{t_2}(\vec{p}) \tag{4}$$

4. Static background: Linear interpolation

$$f_{t_i}(\vec{p}) = W_b \cdot f_{t_1}(\vec{p}) + W_f \cdot f_{t_2}(\vec{p}) \tag{5}$$

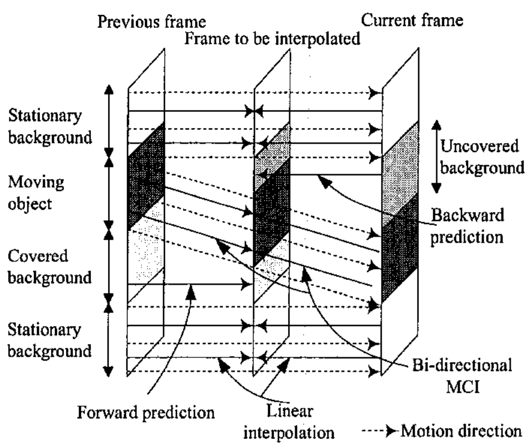


Fig. 1. MCI based on the segmentation.

The MCI method based on the segmentation requires accurate motion estimation for the true motion vector field. Several methods such as 3D-recursive search block matching [6] and hierarchical motion estimation [12] have been proposed to obtain the true motion field. Motion vector refinement techniques based on the spatio-temporal smoothness [13], [14] have also been used to get the true motion vector because the smooth motion vector reflects the true object motion more closely.

Conventional MCI methods produce overlapped pixels and holes in the interpolated frame. This problem is caused by the following reasons: First, although the true motion trajectory is available for a moving object, the object is usually not under rigid translational motion. This case often occurs in sequences with camera motion such as interframe zooming. Second, even if the object is under rigid motion, the estimated motion vector field may not be the same within the same object due to poor motion estimation. In either case, the motion trajectory is not a one-to-one mapping from moving object in the previous frame to that in the current frame. Therefore, the interpolated object tends to contain some overlapped pixels and holes. The mesh-based motion model [3], [15] can avoid this problem since it provides one-to-one mapping between frames. But computational complexity is much higher than the translational motion model, thus mesh-based motion estimation is not adequate in consumer applications.

III. Proposed Frame Rate Up-conversion Using Bi-directional Motion Estimaion

The proposed motion-compensated frame rate up-conversion algorithm uses bi-directional motion estimation to solve the problem of the hole and overlapped regions. The basic principle of the proposed scheme is that although the pixel values in the to-be-interpolated frame are not known, the block motion vectors between the previous and to-be-interpolated frames or between the to-be-interpolated and current frames can be obtained. In video coding such as MPEG, the bi-directional motion vector is used to encode the B (bi-directional) frame. However, the bi-directional motion estimation scheme in video coding cannot be applied directly to the MCI, since, unlike the video coding, pixel values of the to-be-interpolated frame are not known. The proposed method estimates the bi-directional motion vector at each non-overlpping block of the to-be-interpolated frame as shown in Fig. 2. We first estimate the initial motion vector between the available previous and current frames. And then, bi-directional motion estimation is performed using the initial motion vector te be described in the next subsections.

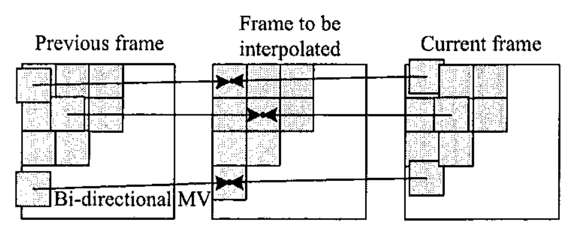


Fig. 2. Vector mapping using the bi-directional motion vector with the non-overlapping contiguous block grid in the frame to be interpolated.

Fig. 3 shows the block diagram of the overall motion-compensated frame rate up-conversion. The proposed scheme is composed of three processing units. First, bi-directional motion estimation block constructs the motion vector field in the to-be-interpolated frame. Second, the motion vector smoothing block smoothes the estimated motion vector in the spatio-temporal direction. Finally, overlapped block motion-compensated interpolation is performed to reduce blocking artifact of the block-based motion estimation and MCI. In the following subsections, we describe each functinal block in detail.

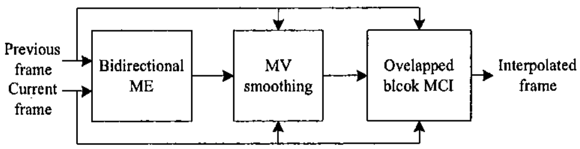


Fig. 3. Proposed frame rate up-conversion scheme.

A. Bi-directional motion estimation in the frame to be interpolated

Fig. 4 illustrates the proposed bi-directional motion estimation scheme. Each frame is first subsampled before initial motion estimation to reduce the computational complexity and to obtain smoothed motion vectors. For initial motion estimation, the full search block-matching algorithm is used in the subsampled images. The estimated motion vector is used to initialize the initial value of the bi-directional motion vector without any modification.

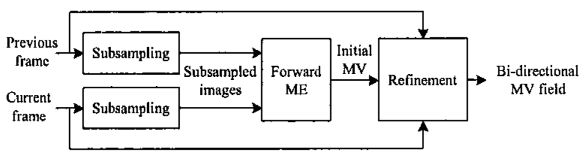


Fig. 4. Bi-directional motion estimation to construct the motion vector field in the to-be-interpolated frame.

In the next step, the initial motion vector is refined using the bi-directional motion estimation with a small search range in the full-scale image. Fig. 5 depicts the refinement process. First, consider the block B_i centered at $\vec{p}$ in the to-be-interpolated frame f_i between the frame f_1 and f_2. With the initial motion vector $\vec{D}_0(B_i)$, we search for the best linear motion trajectory passing through B_i using block-matching. The search range for B_1 in f_1 and B_2 in f_2 is confined to a small displacement $\pm d$ around the initial block position. The initial positions of the center pixels of blocks B_1 and B_2, respectively are calculated using the initial motion vector $\vec{D}_0(B_i)$ as follows

$$(x_1, y_1) = \vec{p} - \vec{D}_0(B_i), \tag{6}$$

$$(x_2, y_2) = \vec{p} + \vec{D}_0(B_i), \tag{7}$$

where $\vec{p} = (x_1, y_1)$ is the center position of the block B_i. Since the two compared blocks B_1 and B_2 moves simultaneously, the number of block matching becomes $(2d+1)^2 - 1$.

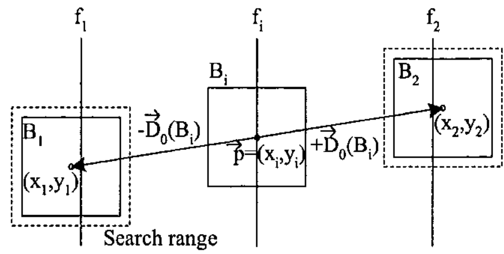


Fig. 5. Refinement of the initial motion vector using bi-directional motion estimation.

B. Spatio-temporal smoothness on the bi-directional motion vector

Once the bi-directional motion vector field is constructed, motion-compensated interpolation is performed to fill the intermediate frame. However, it is often observed that inconsistencies or non-smoothness in the estimated vector field decreases the interpolated picture quality severely. Inconsistencies of the motion vector can be corrected by constraining the spatio-temporal smoothness on the motion vector field.

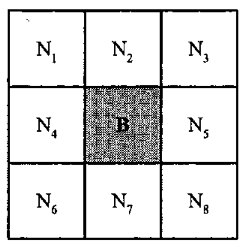


Fig. 6. Neighboring blocks used for motion vector refinement to guarantee the spatio-temporal smoothness.

Next we present the proposed smoothing scheme: Let B and N_i, where $i = 1, 2, ..., 8$ denote the current block and the eight nearest neighboring blocks around B in the to-be-interpolated frame as shown in Fig. 6, and $\vec{D}(B)$ and $\vec{D}(N_i)$ denote the corresponding bi-directional motion vector of B and N_i, respectively. Now, the DFD by the bi-directional motion vector $\vec{D}$ for the current block B can be defined by

$$DFD(\vec{D}, B) = \sum_{\vec{p} \in B} |f_1(\vec{p} - \vec{D}) - f_2(\vec{p} + \vec{D})|. \tag{8}$$

Then, $\vec{D}_s(B)$, the spatio-temporally smoothed motion vector for B is obtained by

$$\vec{D}_s(B) = \arg\min_{\vec{D}} DFD(\vec{D}, B), \tag{9}$$

where $\vec{D} \in \{\vec{D}(B), \vec{D}(N_i)\}$.

The proposed smoothing algorithm is effective in the image boundary; if the object movement occurs near the image boundary, it is very likely to find the incorrect motion vectors since the search range exceeds the image boundary. This problem frequently occurs in sequences with camera motion such as zooming and panning.

C. Overlapped block MCI

In order to construct the intermediate frame, inter-frame interpolation is performed using the motion vectors. In order to interpolate a block B in the to-be-interpolated frame f_i, straightforward block-based bi-directional motion-compensated averaging can be used as follows:

$$f_o(\vec{p}) = \frac{1}{2}[f_1(\vec{p} - \vec{D}(B)) + f_2(\vec{p} + \vec{D}(B))], \qquad (10)$$

where $\vec{p} \in B$ is the pixels in the to-be-interpolated block B. In the same manner, all pixel values in the frame f_i can be simply calculated on a block basis using (10).

The main shortcoming of the above straightforward block-based MCI is the blocking artifact. It is often observed in the constructed frame when motion vectors are not correct or vectors in the neighborhood are significantly uncorrelated. In video coding, the OBMC technique is used to reduce the blocking artifact [16]. This technique, however, cannot be used in the conventional block-based MCI because the block grid in the interpolated frame is not contiguous due to the overlapped pixels and hole regions. Since the proposed frame rate up-conversion scheme produces the interpolated frame with non-overlapping contiguous block grid, the OBMC scheme can be incorporated with our MCI method.

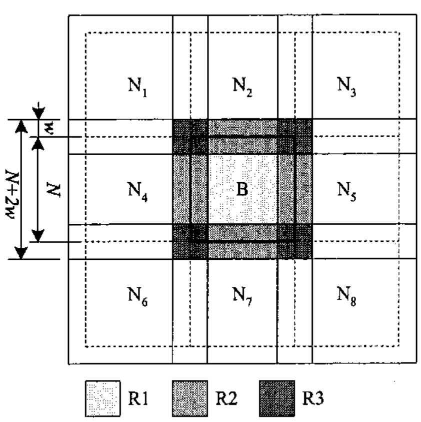


Fig. 7. Block overlapping pattern in the overlapped block MCI.

The proposed MCI using the OBMC scheme employs a simple average interpolation method to reduce the computational complexity. For a given block B with $N \times N$ block size and a small overlapping width w, the original block size is extended to $(N + 2w) \times (N + 2w)$. Since the eight nearest neighboring blocks $N_i, i = 1, 2, ..., 8$, are also extended with the same size, three distinct overlapping regions $R1, R2$, and $R3$ as shown in Fig. 7 are generated. Let $f_o(\vec{p}, \vec{D})$ denote the motion-compensated averaging at $\vec{p}$ using the bi-directional motion vector $\vec{D}$. Then, the output of the overlapped block MCI in the extended block B is defined according to the number of block overlapping as follows:

1. For $R1$: No overlapping

$$f_o(\vec{p} \in R1, \vec{D}(B)) \qquad (11)$$

2. For $R2$:

$$\frac{1}{2}\{f_o(\vec{p} \in R2, \vec{D}(B)) + f_o(\vec{p} \in R2, \vec{D}(N_i))\}, \qquad (12)$$

where $N_i \in \{N_2, N_4, N_5, N_7\}$.

3. For $R3$:

$$\frac{1}{4}\{f_o(\vec{p} \in R3, \vec{D}(B)) + S_k\}, \text{for } k = 1, 2, 3, 4, \qquad (13)$$

where S_k is the sum of the motion-compensated averaging for the neighboring blocks overlapped with B in R3 and defined by

$$\begin{aligned} S_1 &= f_o(\vec{p}, \vec{D}(N_1)) + f_o(\vec{p}, \vec{D}(N_2)) + f_o(\vec{p}, \vec{D}(N_4)), \\ S_2 &= f_o(\vec{p}, \vec{D}(N_2)) + f_o(\vec{p}, \vec{D}(N_3)) + f_o(\vec{p}, \vec{D}(N_5)), \\ S_3 &= f_o(\vec{p}, \vec{D}(N_4)) + f_o(\vec{p}, \vec{D}(N_6)) + f_o(\vec{p}, \vec{D}(N_7)), \\ S_4 &= f_o(\vec{p}, \vec{D}(N_5)) + f_o(\vec{p}, \vec{D}(N_7)) + f_o(\vec{p}, \vec{D}(N_8)). \end{aligned}$$

IV. Experimental Results

In this section, we illustrate some experimental results of the proposed algorithm. Four test sequences are used: `Suzie`, `Flower garden`, `Table tennis`, and `Beach`. Each sequence contains frames with a specific camera motion. Table I summarizes the test sequences and their characteristics.

TABLE I
FOUR TEST SEQUENCES.

Sequence	Frame size	Frame number	Typical camera motion
Suzie	176x144	150	No motion
Flower garden	352x240	115	Panning
Beach	352x240	184	Gradual scene change
Table tennis	352x240	150	Zooming and Abrupt scene change

The odd frames of these test sequences are eliminated for simulations. Conventional MCI algorithm consists of motion estimation using full search block matching, the refinement for spatial smoothness in [6], covered/uncovered classification in [11] for hole regions, and method in [17] for overlapped pixels. For each sequence, we perform 1:2 frame rate up-conversion and compute the average PSNR between the original and interpolated frames. The block size is 16×16 and the search range is ±16. The search range is from -2 to 2 in the motion estimation for the refinement of the initial motion vector. In overlapped block MCI, we use the overlapping width $w = 2$.

A. Comparison with conventional MCI

Fig. 8 shows the MCI results of the proposed algorithm for the `Table Tennis` sequence in zoom motion. Since the conventional method produces overlapped and hole regions as shown in Fig. 8(b), the interpolated image has annoying artifact along the edge of the table (see Fig. 8(c)). The proposed scheme shows good result in that area as shown in Fig 8(e). The values in the error image in Fig. 8 (d) and (f) are multiplied by a factor of 4 for the better visual inspection.

Table II summarizes the simulation results of each test sequence. From the results, we can see that the performance enhancement of the sequences with camera motion is much more than that of `Suzie` sequence with no camera motion. Consequently, we can conclude that the proposed algorithm exhibits a better performance especially in sequence with camera motions. Note that, in the `Table tennis` sequence, abrupt scene change occurs two times. In that case, motion-compensated interpolation is switched off, and simple frame repetition is used.

TABLE II
PSNR COMPARISON.

Sequence	Conventional MCI	Proposed algorithm	Gain (dB)
Suzie	40.33	40.93	+0.60
Flower garden	29.38	30.84	+1.46
Beach	35.88	36.92	+1.04
Table tennis	27.22	29.07	+1.85

B. Effectiveness of each functional block in the proposed scheme

We performed further experiments to verify the effectiveness of each functional block in the proposed algorithm. The tested function blocks are

(a) Refinement of initial motion vector,
(b) Spatio-temporal smoothing, and
(c) Overlapped block MCI.

In order to investigate how each block affects on the interpolated image quality, we simulated four types of frame rate up-conversion,

(1) without any of them, only with initial motion vector,
(2) with (a) only,
(3) with (a) + (b), and
(4) with all functions, (a) + (b) + (c).

Fig. 9 shows that the PSNR performance is improved as a functional block is added, i.e., the type number increases. Note the effects of algorithm (b) in the result of type (3) for the sequence `Flower garden`, where the motion vector is highly correlated spatially and temporally since the sequence is in panning camera motion. Therefore, the spatio-temporal smoothing function performs well for this sequence.

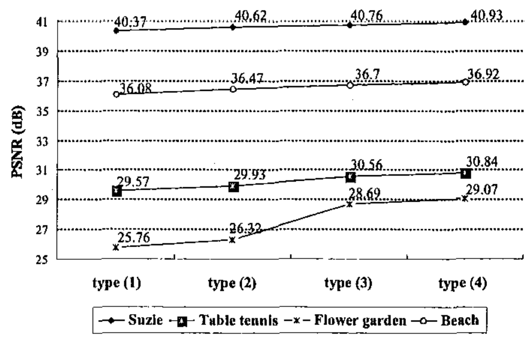


Fig. 9. Effects of each functional block on the overall performance.

V. Conclusions

In this paper, a new motion-compensated interpolation algorithm for frame rate up-conversion using bi-directional motion estimation has been proposed. The proposed scheme is composed of three functional units; bi-directional motion estimation, spatio-temporal smoothing, and overlapped block motion compensated interpolation. The main feature of the proposed motion-compensated frame rate up-conversion scheme is that, unlike conventional MCI algorithms, the proposed technique does not produce any overlapped pixel and hole region in the interpolated frame. Moreover, by using the overlapped block motion compensation technique, the blocking artifact of the block-based motion estimation is effectively eliminated. It is very simple to implement the proposed MCI algorithm on consumer products when compared to conventional MCI methods due to its simplicity. Experimental results has proven that the proposed algorithm has better performance than the conventional MCI algorithm and is very robust especially in sequences with various camera motions like panning and zoom.

Acknowledgements

This work was supported by Samsung Electronics. Co., Korea.

References

[1] E. Petajan, "The HDTV Grand Alliance System," in *Proc. IEEE*, vol. 83, pp. 1094–1105, Jul. 1995.
[2] K. A. Bugwadia, E. D. Petajan, and N. N. Puri, "Progressive-Scan Rate Up-Conversion of 24/30 Source Materials for HDTV,"

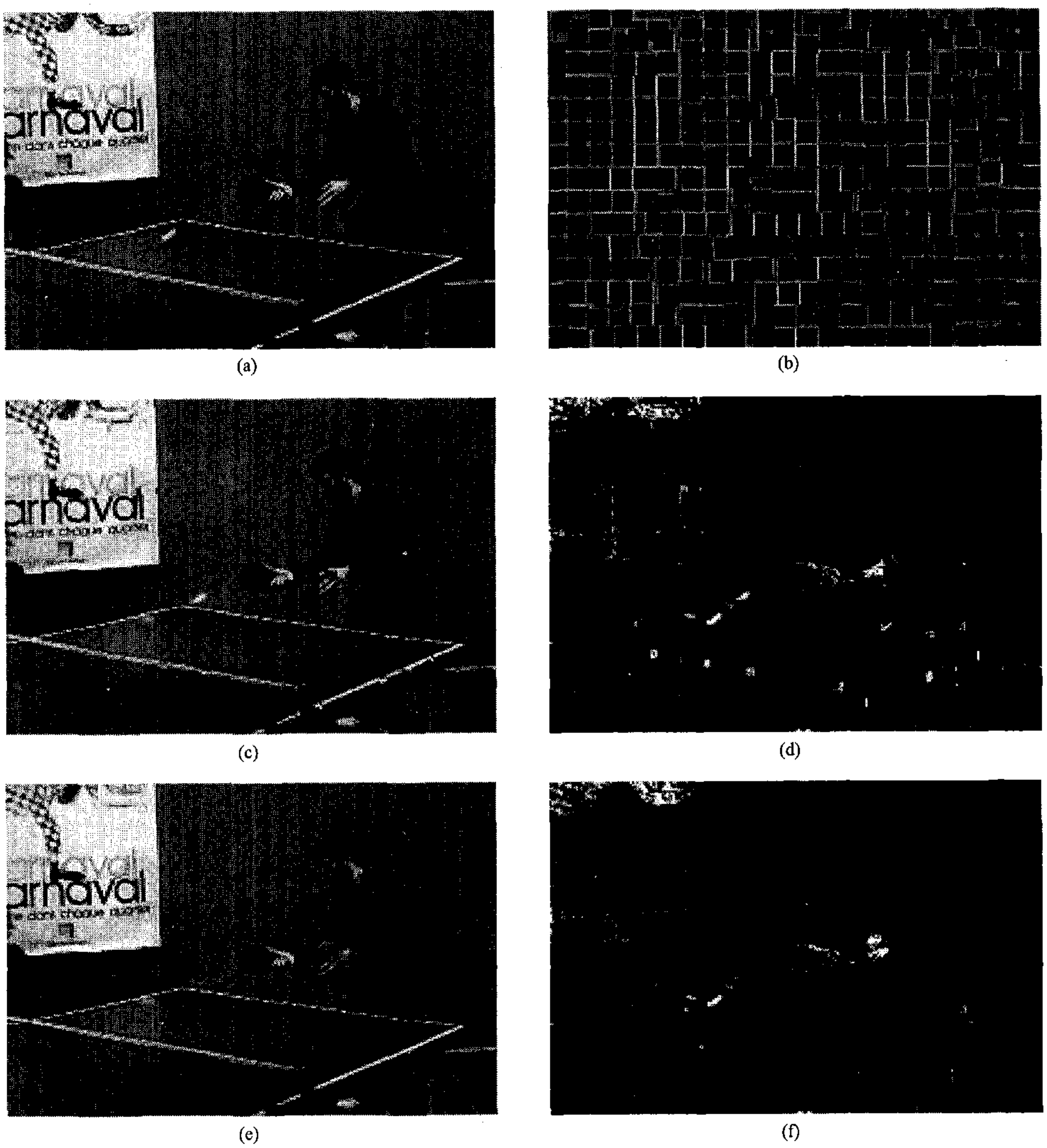


Fig. 8. Test sequence `Table Tennis`: (a) Original 81th frame, (b) overlapped and hole regions in the interpolated frame using the conventential MCI (c) interpolated image using the conventional MCI, (d) error image between (a) and (b), (e) interpolated image using the proposed algorithm, and (f) error image between (a) and (e).

IEEE Trans. Consumer Electronics, vol. 42, no. 3, pp. 312–321, Aug. 1996.

[3] 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*, vol. 1, pp. 747–750, Oct. 1997.

[4] R. Thoma and M. Bierling, "Motion compensating interpolation considering covered and uncovered background," *Signal Processing: Image Compression*, vol. 1, pp. 192–212, 1989.

[5] J. N. Youn, M. T. Sun, and C. W. Lin, "Motion estimation for high performance transcoding," *IEEE Trans. Consumer Electronics*, vol. 44, no. 3, pp. 649–658, 1998.

[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 Trans. Circuits Syst. Video Technol.*, vol. 3, no. 5, pp. 368–379, Oct. 1993.

[7] P. Moulin, R. Krishnamurthy, and J. W. Woods, "Multiscale modeling and estimation of motion fields for video coding," *IEEE Trans. Image Processing*, vol. 6, no. 12, pp. 1606–1620, Dec. 1997.

[8] T. Y. Kuo and C. -C. J. Kuo, "Motion-compensated interpolation for low-bit-rate video quality enhancement," in *Proc. SPIE Visual Communications and Image Processing*, vol. 3460, pp. 277–288, July 1998.

[9] A. Kaup and T. Aach, "Efficient prediction of uncovered background in interframe coding using spatial extrapolation," in *Proc. ICASSP.*, vol. 5, pp. 501–504, 1994.

[10] R. J. Schutten and G. D. Haan, "Real-time 2-3 pull-down elimination applying motion estimation/compensation in a programmable device," *IEEE Trans. Consumer Electronics*, vol. 5, pp. 501–504, 1994.

[11] P. Csillag and L. Boroczky, "Enhancement of video data using motion-compensated postprocessing techniques,"in *Proc. ICASSP.*, vol. 4, pp. 2897–2900, 1997.

[12] M. Bierling and R. Thoma, "Motion Compensating field interpolation using a hierarchically structured displacement estimator," *Signal Processing*, vol. 11, No. 4, pp.387-404, Dec. 1986.

[13] Y. K. Chen, A. Vetro, H. Sun, and S. Y. Kung, "Frame-rate upconversion using transmitted true motion vectors,"in *Proc. IEEE Second Workshop on Multimedia Signal Processing*, pp. 622-627, 1998.

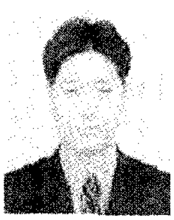
[14] J. N. Youn and M. T. Sung, "A fast motion vector composition method for temporal transcoding," in *Proc. IEEE Int. Symp. Circuits and Systems*, vol. 4, pp. 243-246, May 1999.

[15] F. J. P. Lopes and M. Ghanbari, "Improved motion compensation with overlapped spatial transformation," in *Proc. IEEE Int. Conf. Electronics, Circuits and Systems - Surface the Waves of Science and Technology*, vol. 2, pp. 457-460, Sep. 1998.

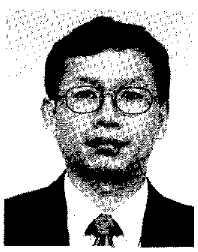
[16] T. Y. kuo and C. -C. J. Kuo, "Complexity reduction for overlapped block motion compensation (OBMC)," in *Proc. SPIE Visual Communication and Image Processing*, vol. 3024, pp. 303–314, Feb. 1997.

[17] D. W. Kim, J. T. Kim, and I. H. Ra, "A new video interpolation technique based on motion-adaptive subsampling," *IEEE Trans. Consumer Electronics*, vol. 45, no. 3, pp. 782–786, Aug. 1999.

Byung-Tae Choi (S'96) was born in Korea, on November 9, 1970. He received the B.S. and M.S. degree in electronic engineering with the Department of Electronic Engineering from Korea University, in 1994, and 1996, respectively. In 1998, he joined the Research Institute for Information and Communication Technology, Korea University, Korea, as a researcher. His research interests are in the areas of video signal processing and multimedia communications.



Sung-Hee Lee (M'00) received the Ph.D. degree in Electronics Engineering and the M.S. degree in Computer Science from Korea University in 2000 and 1995, respectively, and the B.S. degree in Information Engineering at Korea University in 1993. He is now a research engineer in Samsung Electronics Co., Ltd. His research interests are in the areas of multimedia communications and nonlinear signal, and image processing.



Sung-Jea Ko (M'88-SM'97) received the Ph.D. degree in 1988 and the M.S. degree in 1986, both in Electrical and Computer Engineering, from State University of New York at Buffalo, and the B.S. degree in Electronic Engineering at Korea University in 1980. In 1992, he joined the Department of Electronic Engineering at Korea University where he is currently a Professor. From 1988 to 1992, he was an Assistant Professor of Electrical and Computer Engineering at the University of Michigan-Dearborn. From 1981 to 1983, he was with Daewoo Telecom Corporation where he was involved in research and development on data communication systems. He received the Hae-Dong best paper award from the Institute of Electronics Engineers of Korea (1997), and the best paper award from the IEEE Asia Pacific Conference on Circuits and Systems (1996). He is the 1999 Recipient of the LG Research Award given to the Outstanding Information and Communication Researcher. Dr. Ko is currently the Consumer Electronics chapter chairman of the IEEE Korea Council. His current research interests are in the areas of digital signal and image processing, and multimedia communications.