LETTER

Video Frame Interpolation by Image Morphing Including Fully Automatic Correspondence Setting

Miki HASEYAMA[†a)], ***Member***, **Makoto TAKIZAWA**[†], ***and*** **Takashi YAMAMOTO**[†], ***Student Members***

SUMMARY In this paper, a new video frame interpolation method based on image morphing for frame rate up-conversion is proposed. In this method, image features are extracted by Scale-Invariant Feature Transform in each frame, and their correspondence in two contiguous frames is then computed separately in foreground and background regions. By using the above two functions, the proposed method accurately generates interpolation frames and thus achieves frame rate up-conversion.

key words: *video frame interpolation, image morphing, global motion estimation, scale-invariant feature transform*

1. Introduction

Several methods for interpolation of video frames have recently been proposed [1]–[3]. In these methods, correspondence of pixels between two contiguous frames is estimated by using motion vectors. According to the correspondence, the intensities of interpolated frames are obtained. However, since this approach assumes that neighboring pixels have similar motion vectors, the interpolation results sometimes contain artifacts, especially near boundaries between regions whose motions are different. Actually, in the interpolation results, over-smoothing appears in the boundaries.

A method to solve this problem is proposed here. The method is based on image morphing [4], which deals with the metamorphosis of one image to another and is known as image interpolation in the time domain. Based on this property, it is reasonable to expect that image morphing can be used for frame rate up-conversion. However, image morphing cannot be simply applied to video frame interpolation because it generally requires the feature correspondence between the two images via a user interface, which is a critical drawback to application. To overcome this drawback, feature must be automatically obtained, and a method for automatically obtaining feature correspondence by using Scale-Invariant Feature Transform (SIFT) [5] is proposed here. Furthermore, in order to obtain high-quality interpolation frames, especially in the boundaries, global motion estimation [6] is utilized as a preprocess.

This letter is organized as follows. In Sect. 2, image morphing and global motion estimation, which are used in the proposed method, are simply introduced. In Sect. 3, the proposed method is explained in detail. Finally, experimental results are shown in Sect. 4 to verify the high performance of the proposed method.

2. Preparation —— Image Morphing and Global Motion Estimation ——

In this section, the image morphing method and the global motion estimation method, which are used in the proposed method, are simply explained.

2.1 Image Morphing

Image morphing generates continuous transformation of one image into another. These two images are called target images, and the images generated continuously between the target images are called middle images. Field morphing [4] is a well-known image morphing method and is used in our method. In field morphing, a certain number of lines, which are called control lines, must be set, and field morphing is executed by the following two functions: 1) warping, in which coordinates of the control lines in the middle images are calculated, and 2) cross dissolving, in which intensities of the pixels in the middle images are calculated. Before executing the above functions, correspondence of the control lines between the target images must be obtained via a user interface. However, as stated in Sect. 1, this is a critical drawback to application. Thus, in order to automatically obtain the correspondence between two images, control "points" instead of control lines are adopted in the proposed method. By this modification, the correspondence can be obtained automatically without any requirement via the user interface.

2.2 Global Motion Estimation Method

Global motion is the motion in the background, which is usually due to camera movement in a video sequence. Global motion estimation proposed in [6] consists of three stages, and the following two stages are used in the proposed method:

- First stage
  Moving objects are detected and negative effect on the following global motion estimation is eliminated.
- Second stage
  Global motion is estimated, and the model is represented by the projective parameters a_i $(i = 1, \ldots, 8)$

Manuscript received March 27, 2009.

[†] The authors are with the Graduate School of Information Science and Technology, Hokkaido University, Sapporo-shi, 060–0814 Japan.

a) E-mail: miki@ist.hokudai.ac.jp

DOI: 10.1587/transinf.E92.D.2163

as follows:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \frac{\begin{pmatrix} a_1 & a_2 \\ a_4 & a_5 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a_3 \\ a_6 \end{pmatrix}}{\begin{pmatrix} a_7 & a_8 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}}, \tag{1}$$

where the coordinate (x', y') on the previous frame is transformed to (x, y) on the present frame.

The first stage is utilized for the proposed method to separate the foreground and background regions in each frame. Based on the global motion obtained in the second stage, the proposed method computes the correspondence of the pixels in the background between the two contiguous frames.

3. Image Morphing-Based Video Frame Interpolation

The proposed method generates interpolation frames between two contiguous frames in a given video sequence. They are the current frame I_t and the previous frame I_{t-1}, and $I_t(x, y)$ and $I_{t-1}(x', y')$ are the intensities in the coordinate (x, y) of I_t and the coordinate (x', y') of I_{t-1}, respectively. The interpolated frames are generated as follows.

Preprocessing: Region segmentation based on global motion
The global motion estimation method shown in 2.2 is applied to the frames I_t and I_{t-1} and the global motion parameters are obtained. According to the obtained global motion parameters, the coordinate (x', y') on I_{t-1} is transformed to (x, y), and the following flag image I_t^F is generated as follows:

$$I_t^F(x, y) = \begin{cases} 1 & \text{if } |I_t(x, y) - I_{t-1}(x', y')| \geq C_{Th} \\ 0 & \text{otherwise} \end{cases}. \tag{2}$$

In the above equation, if $I_t^F(x, y) = 1$, the pixel in (x, y) of I_t belongs to the foreground regions; otherwise the pixel (x, y) belongs to the background regions.

Procedure 1: Automatic Setting of Control Points
The image features in I_t and I_{t-1} are extracted by [5] and used as the control points in image morphing.

Procedure 2: Video Frame Interpolation by Image Morphing
The correspondence between features in I_t and I_{t-1} is computed by the method in [5], which is executed independently in each of the foreground and background regions obtained by Preprocessing. Then, based on the correspondence, interpolated frames are obtained as the middle images generated by image morphing.

Procedure 1 and Procedure 2 are described in detail in the following subsections.

3.1 Procedure 1 —— Automatic Setting of Control Point by SIFT ——

Procedure 1 sets the control points in the two contiguous frames I_{t-1} and I_t, where the total number of frames is T, as follows.

Step 1: Feature Point Extraction
The feature points $F_{t-1}(j)$ and $F_t(k)$ ($j = 1, 2, \cdots, M$; $k = 1, 2, \cdots, N$) are detected by [5] in the frames I_{t-1} and I_t, respectively, where M and N are the total numbers of feature points extracted in I_{t-1} and I_t, respectively.

Step 2: Computation of Image Feature Correspondence
The best correspondence feature point with $F_{t-1}(j)$ is selected among $F_t(k)$ $k = 1, 2, \cdots, N$), according to the criterion in [5], which is defined as the distance between two different feature vectors. However, in our method, only the corresponding points satisfying the following equation remain to be processed in Procedure 2:

$$\frac{D_{first}}{D_{second}} < Th. \tag{3}$$

In the above equation, Th is a predefined threshold, D_{first} is the distance between the feature vector of $F_t(k)$ and the feature vector of the best matched feature point in the frame I_{t-1}, and D_{second} is the distance between the feature vector of $F_t(k)$ and the second best matched feature vector. This matching strategy shown in Eq. (3) is commonly used, such as shown in [7].

Step 3: Selection of Blank Blocks
The current frame I_t is partitioned into blocks with sizes of $S \times S$ pixels. All of the blocks that do not include any control points are selected. These blocks are called blank blocks hereafter.

3.2 Procedure 2 —— Video Frame Interpolation by Image Morphing ——

Video frame interpolation in the foreground and background regions is achieved by the proposed method. The details are given below.

(i) Foreground regions
The intensities of pixels in the interpolated frames that are originally located in the foreground region are calculated by using image morphing. For execution of image morphing, the correspondence of feature points is obtained by Step 1 in 3.1. However, the intensities in the blank blocks, which are obtained by Step 3 in 3.1, are computed by [3].

(ii) Background regions
The intensities of pixels in the interpolated frames that belong to the background regions obtained by Preprocessing are computed as follows. First, according to

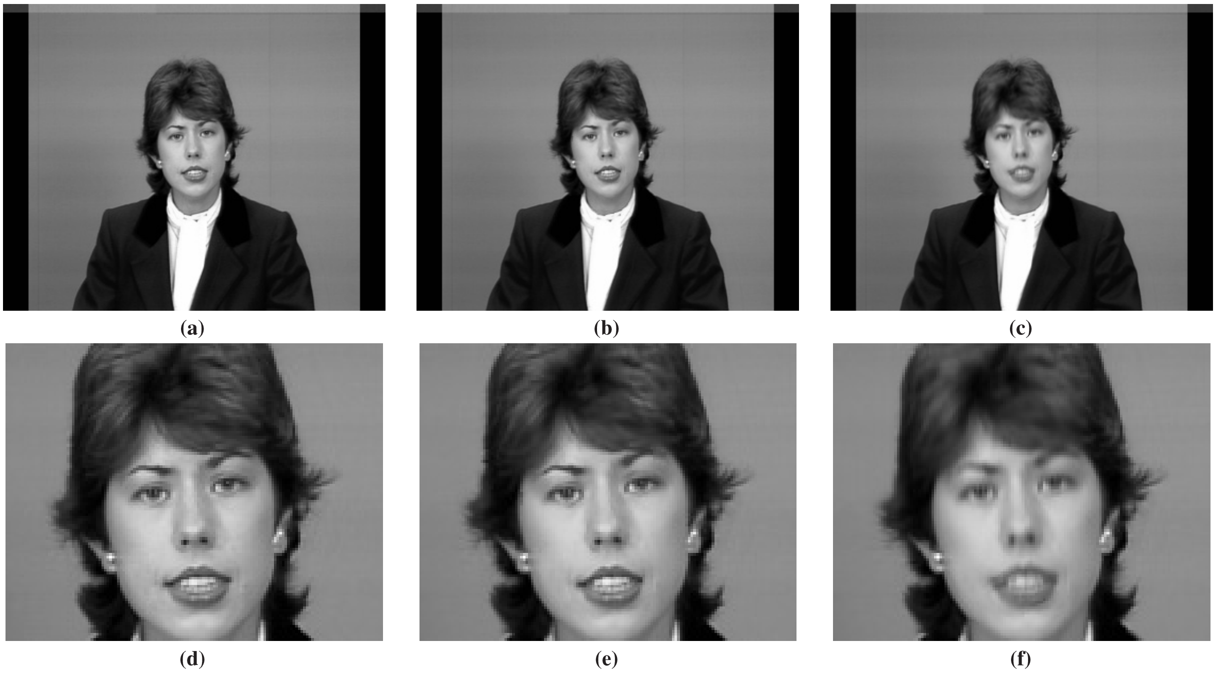

Fig. 1 Experimental results I: (a) original frame of *Claire*, (b) result of interpolation by the proposed method, (c) result of interpolation by [3], (d) enlarged image of (a), (e) Enlarged image of (b), (f) Enlarged image of (c).

the estimated global motion, which can be considered as the correspondence between the pixels in I_t and I_{t-1}, the proposed method warps each pixel in the background regions of I_{t-1} to a pixel in the background regions in I_t and finally generates background regions of in-between frames by cross-dissolving of the intensities at each corresponding pair of pixels.

Since the proposed method separately interpolates the foreground and background regions, sharp edges of the objects in the foreground region are retained. Also, in the background regions, the interpolation quality is less affected by the motion of foreground objects than the interpolation quality when it is not used.

4. Experimental Results

Experimental results verifying performance of the proposed method are presented in this section. We used a test video sequence *Claire* of 360 × 288 pixels, 8 bits/pixel and 30 fps. In order to obtain a low-frame-rate video sequence, we subsampled it to 7.5 fps. Then we applied the proposed method to the low-rate sequence and generated a frame rate up-converted video sequence at 15 fps using $C_{Th} = 25$ in Eq. (2), $Th = 0.1$ in Eq. (3), and $S = 24$ in Step 3 of Subsection 3.1. These parameters are empirically set.

For subjective evaluation, the interpolated frames are shown in Fig. 1: (a) is an original frame of *Claire*, (b) is the result of interpolation by the proposed method and, (c) is the result by [3]. Enlarged portions around the face in *Claire* are shown in Fig. 1 (d)–(f). From these figures, we can see that sharper edges than those in [3] are obtained by using the proposed method.

The proposed method was also applied to another video sequence, *City*, of 352 × 288 pixels, 8 bits/pixels, 30 fps, and total number of frames of 265. For the experiment, we subsampled the original sequence to 5 fps, and then we applied the proposed method to the low-rate sequence and obtained the interpolated video sequence of 10 fps. Finally, the total number of interpolation frames was 89, including 45 original frames.

From the results, the average accuracy of all of the frames is 27.20 dB. For comparison, a conventional method [3] was also applied to the same sequence, and its accuracy was 26.14 dB. An improvement in accuracy of 1.06 dB was achieved by using the proposed method. The PSNR between the original frame and its interpolation result is shown in Fig. 2 (a). In the results obtained by using the proposed method, there are some low-accuracy interpolation frames, such as the 32nd and 76th frames. This is caused by failure of correspondence estimation and can be improved by using estimation results of multiple frames. Figures (b)–(d) are also shown for subjective comparison of these results. Based on these figures, we can see that not only more accurate interpolation than that in [3] but also that sharper edges, especially in the building in the center of the frame, are obtained.

5. Conclusions

A video frame interpolation method using the image morphing is proposed. Feature correspondence between two con-

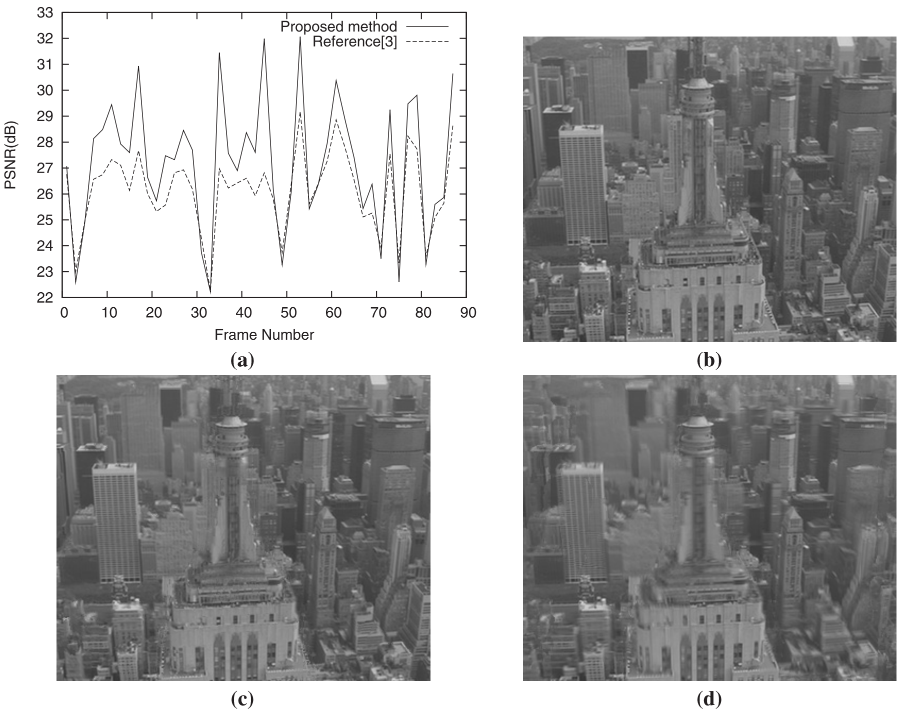

Fig. 2 Experimental Results II: (a) PSNR between the original sequence *City* and the result of interpolation by the proposed method, (b) original frame corresponding to the 46th interpolation frame at 10 fps, (c) result of interpolation by the proposed method, and (d) result of interpolation by [3].

tiguous frames based on SIFT is automatically obtained by the proposed method. In order to generate high-quality interpolation frames, the proposed method separately executes interpolation in the foreground and background regions. Experimental results verified that the proposed method preserves sharp edges.

Acknowledgment

This research was partially supported by the Global COE project in the Graduate School of Information Science and Technology, Hokkaido University.

References

[1] R. Castagno, P. Haavisto, and G. Ramponi, "A method for motion adaptive frame rate up-conversion," IEEE Trans. Circuits Syst. Video Technol., vol.6, no.5, pp.436–446, 1996.

[2] T.-Y. Kuo, J. Kim, and C.-C.J. Kuo, "Motion-compensated frame interpolation scheme for H.263 codec," Proc. IEEE Int. Symp. Circuits and Systems(ISCAS '99), vol.4, pp.491–494, Orlando, Fla, USA, May-June 1999.

[3] H.A. Karim, M. Bister, and M.U. Siddiqi, "Multiresolution motion estimation for low-rate video frame interpolation," EURASIP J. Applied Signal Processing, vol.2004, no.11, pp.1708–1720, 2004.

[4] T. Beier and S. Neely, "Feature-based image metamorphosis," SIGGRAPH 92 Conference Proceedings, pp.35–42, 1992.

[5] L. Lowe, "Distinctive image features from scale invariant keypoints," Proc. International Journal of Computer Vision (IJCV), vol.60, no.2, pp.91–110, 2004.

[6] C.T. Hsu and Y.C. Tsan, "Mosaics of video sequences with moving objects," Signal Process., Image Commun., vol.19, no.1, pp.81–98, 2004.

[7] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," IEEE Trans. Pattern Anal. Mach. Intell., vol.6, no.5, pp.436–446, 1996.