Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Exemplar-based video inpainting with large patches

Abbas KOOCHARI, Mohsen SORYANI

(*Department of Computer Engineering, Iran University of Science and Technology, Tehran 16846-13114, Iran*)

E-mail: {koochari, soryani}@iust.ac.ir

Received May 24, 2009; Revision accepted Oct. 28, 2009; Crosschecked Mar. 1, 2010

Abstract: Inpainting is the process of reconstructing damaged regions of images and video frames. This study deals with weaknesses of the current video inpainting techniques, when an object is totally damaged, and a framework for video inpainting is proposed. Using this framework, the moving object is separated from the background. A large mosaic image is constructed using the moving object and then a patch-based method with large patches is used to fill holes. In each frame, the inpainted foreground is obtained by placing the object in its location. Missing areas of the stationary background are also filled separately and the final video is produced by composing the inpainted background and object frames. Results for three video sequences with an occluded object show that this approach represents the object in the missing region better than other approaches.

Key words: Video inpainting, Patch, Background modeling, Mosaic
doi:10.1631/jzus.C0910308 **Document code:** A **CLC number:** TP391.41

1 Introduction

Image and video inpainting are two interesting areas in image processing which have drawn much attention in the past few years. There have been a large variety of applications such as image retouching, image compression, and video editing (Zhang *et al*., 2005; Matsushita *et al*., 2006; Liu *et al*., 2007). The main focus of image and video inpainting is to fill in the missing parts of a frame, caused by image scratches, hand manipulations, or block loss in data transmission, using information from the target frame or its adjacent frames (in case of a video sequence), such that the inpainting is undetectable.

Several works have been performed on digital image inpainting (Bertalmio *et al*., 2000; 2003; Oliveira *et al*., 2001; Sun *et al*., 2005; Ho and Goecke, 2007), most of which are based on extending surrounding pixels of the missing part until that part fades away. Criminisi *et al*. (2004) proposed a priority criterion for selecting the surrounding pixels to fill large holes.

Temporal continuity of an object's motion should be preserved in video sequences; however, none of these algorithms can be applied directly to video inpainting. The work of Bertalmio *et al*. (2001) might be the first attempt to address this problem, in which the partial difference equation method was applied to all frames of a video sequence to maintain continuity. Wexler *et al.* (2004; 2007) framed the space time video completion task as a global optimization problem with well-defined local and global constraints. Zhang *et al*. (2005) proposed a video completion scheme based on motion layer estimation and segmentation where motion compensation was used to complete the moving object. To complete video for a perspective camera, Shen *et al.* (2006) separated the foreground from the background, constructed manifolds of space-time volume, and then rectified the object volume to repair perspective distortion; the output of this approach is acceptable, but it has artifacts and does not work well for large missing regions. Patwardhan *et al.* (2007) separated the moving foreground from the background in a pre-processing step, filled the missing data as much as possible by information from the moving foreground

of other frames, and inpainted the remaining from the local background. Although the overall performance of this approach is good, it does not work well when the moving object is near or enters a large missing region. To inpaint moving objects, Cheung *et al.* (2006) extracted a set of object templates and utilized a dynamic programming technique for optimal object manipulation. The drawback of this approach is that a jump is incurred when entering and leaving the hole. Wang *et al.* (2007) proposed a feature-based video inpainting technique for largely occluded moving human subjects which models human behavior with predefined features.

This paper deals with a framework for filling missing parts in video frames in the presence of an occluding object. The background is assumed to be stationary and motion of the moving foreground is considered to be periodic without scale changing in the object.

2 The proposed method

We have developed a novel video inpainting approach for filling in the missing data using a patch-based inpainting approach proposed by Criminisi *et al*. (2004). In the current research, some weaknesses of the current video inpainting techniques, when an object is totally damaged, are mentioned and a framework for video inpainting is proposed. Fig. 1 shows a schematic overview of the algorithm. First, the moving foreground is separated from the stationary background using a simple thresholding mechanism; then each segment is inpainted separately; and finally the video is reconstructed by composing the inpainted background and foreground frames.

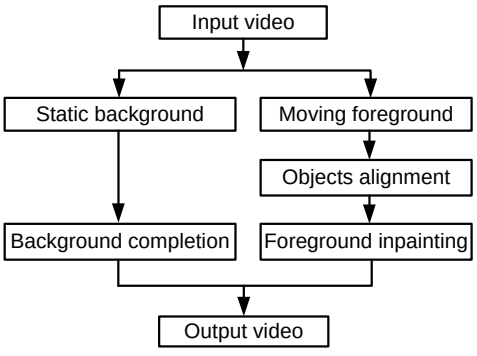

Fig. 1 Schematic overview of the proposed algorithm

2.1 Separating the moving foreground from the background

In order to discriminate the moving object from the background in each frame, we used a threshold value on the absolute difference between the frame and the modeled background (Eq. (1)).

$$\mathrm{FG}_t = \begin{cases} 1, & \text{if } \left|I_t - \mathrm{BG}_{t-1}\right| > \mathrm{thr}, \\ 0, & \text{otherwise}, \end{cases} \tag{1}$$

where I_t is the input frame at time t, BG_{t-1} is the background model, and FG_t is the result of foreground at frame t. For background modeling, the Gaussian mixture model (GMM), originally introduced in (Wren *et al.*, 1997; Stauffer and Grimson, 1999) and improved in Zivkovic (2004), is used. It models each pixel as a mixture of k Gaussian distributions and constantly updates the parameters of the model. The number of components of GMM is automatically selected for each pixel. Given the observation of a pixel at previous time, the probability of observing the pixel at time t is

$$P(X_t) = \sum_{i=1}^{k} w_{i,t} N(X_t, \mu_{i,t}, \boldsymbol{\Sigma}_{i,t}), \tag{2}$$

where k is the number of components, and $w_{i,t}$, $\mu_{i,t}$, and $\boldsymbol{\Sigma}_{i,t}$ are the estimation of weight, mean value, and covariance matrix of the ith Gaussian in the mixture at time t, respectively. After modeling each pixel, the background can be approximated by the B largest components of the mixture model where B is defined as

$$B = \arg\min_b \left(\sum_{k=1}^{b} w_k > T \right), \tag{3}$$

where T is the minimum portion of data that can be considered as background. As explained above, the moving foreground is recognized by the difference of each frame from the modeled background. Since the background subtraction is done at the pixel-level, a median filter is applied to each detected foreground frame in order to remove the outlier pixels.

Background subtraction is still a challenging task in computer vision. Since the focus of our work is to develop a better inpainting algorithm and a stationary background is used in our video sequences,

the above-mentioned foreground/back-ground segmentation method is suitable for this work. More advanced foreground/background segmentation methods can be used if required. For dynamic backgrounds, there are other background subtraction methods such as presented in Sheikh and Shah (2005).

2.2 Alignment of the moving objects

After segmenting the moving objects and the background, the boundary of each object is specified by a rectangular window around it. Since deformable moving objects are in different states of motion, the maximum width and height of windows are selected to define a reference window. The reference window encloses the object as the object settles in the center of window. Afterwards, a large mosaic image is obtained by the alignment of windows which contain the moving objects. To remove any possible overlap between objects, the center of each object is defined as follows:

$$\mathrm{cm}_i = \mathrm{cm}_{i-1} + w + \Delta c_i, \quad \Delta c_i = c_i - c_{i-1}, \tag{4}$$

where cm_i is the center of the object at frame i in the mosaic image, w is the width of the reference window, and Δc_i is the distance between object centers at two consecutive frames (i.e., velocity of the object at frame i).

Before mosaic construction, the first and the last damaged frames are manually determined. Since, in the damaged frames, the object center is corrupted, it is estimated using the velocity of the object. Also, all damaged regions of the manually-determined frames are copied onto the mosaic image. Fig. 2 shows the mosaic image of correct frames. Since the mosaic image is large, only a section of it has been shown.

2.3 Background and foreground inpainting

Background and foreground inpainting are done using exemplar-based inpainting (Criminisi *et al*., 2004), which has been shown to work well in images with both texture and structure. For background inpainting we have directly used the exemplar-based method on the background image, while for the foreground inpainting we iteratively apply exemplar-based inpainting to the mosaic image.

2.3.1 Exemplar-based method

The target region (the missing data region) is filled by selecting patches from the source region. The priority of the first area to be filled and patch selection order are determined with respect to the structure of the entire image. The target region, source region, and entire image are denoted as Ω, Φ, and I, respectively. In order to fill a selected region, the priority of each pixel on the boundary of the target region, $\delta\Omega$, is computed. The priority of a pixel p is calculated as follows:

$$P(p)=C(p)D(p), \tag{5}$$

where $C(p)$ and $D(p)$ are confidence and data terms respectively, which are calculated as follows:

$$C(p) = \left(\sum_{q\in\Psi_p\cap\Phi} C(q)\right) \Big/ |\Psi_p|, \tag{6}$$

$$D(p) = |\nabla I_p^{\perp} \cdot \boldsymbol{n}_p| / \alpha, \tag{7}$$

where Ψ_p is a patch centered in location p, $|\Psi_p|$ is the area of the patch, α is the normalization factor (e.g., α=255 for grayscale images), $\boldsymbol{n}_p$ is normal to the boundary of the target region, and $\nabla I_p^{\perp}$ is an isophote in location p (Fig. 3).

The patch $\Psi_{\hat{p}}$ with the maximum priority is found in the target region (i.e., $\Psi_{\hat{p}} \mid \hat{p} = \arg\max_{p\in\partial\Omega} P(p)$), and the best matching patch $\Psi_{\hat{q}}$ in the source region that minimizes the sum square error (SSE) is selected and copied into $\Psi_{\hat{p}}$. At the end, confidence terms for all pixels of the selected patch intersecting with the target region are updated. This algorithm is run iteratively until the target region is filled.

Fig. 2 Mosaic image of the correct frames obtained by objects alignment

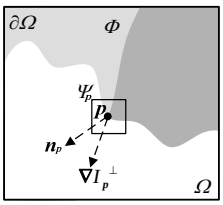

Fig. 3 Notation used in exemplar-based inpainting (Criminisi *et al*., 2004)
Given the block Ψ_p, $\mathbf{n}_p$ is the normal to the contour $\partial\Omega$ of the hole region Ω. Φ is the non-hole region. $\nabla I_p^{\perp}$ is the isophote at point p

2.3.2 Background inpainting

Since the missing region in the static background is the same for all frames, it must be inpainted using the information of the current frame. The exemplar-based approach, which maintains texture and structure information, is the best candidate to fill this region, although there are difficulties in some complex curved structures. In the case of moving backgrounds, a frame history can be used to fill the missing part of the current frame. This is, however, more challenging compared to a stationary background since the temporal discontinuation is more visible in moving backgrounds than in static backgrounds.

2.3.3 Foreground inpainting

The main step in this approach is the completion of missing data in a mosaic image. An exemplar-based image inpainting with large patches is used for the completion of a mosaic image. This step is iteratively run and in each iteration one or two objects are inpainted until the missing region is filled. To simplify the explanation, one moving object has been considered. The patch should be sufficiently large to enclose an object entirely. Selection of a large patch size preserves temporal continuity and structure of the moving object. Also, object motion should be periodic without scale changes for this purpose.

In the proposed approach, the height of the patch is considered equal to the height of the reference window while its width is considered three fold of the width of the reference window. The reason for this assumption is to maintain continuity of the current object with the left and the right objects. Although the selected patch might contain only one of the right or left objects, it preserves periodicity of the sequentially filled moving objects. The missing region can be filled from left or right side of the mosaic image based on priority of patches. When a large patch with the maximum priority is selected to inpaint the mosaic image, it may include an object and a section of another object. Since usually one or two objects should be inpainted entirely in each iteration, a projection is applied to the selected patch to distinguish the boundaries of two neighboring objects. This approach insures that one or two objects are inpainted entirely in each iteration and uncompleted objects are not included in the patch.

If the object sequence is denoted by $o_1o_2...o_co_{c+1}\ldots o_{c+n}o_{c+n+1}\ldots o_N$, and T is period of the objects' movement, we have $o_i \cong o_{i+T}$ (i is the frame number, and o_c to o_{c+n} are objects in the corrupted region). When a patch is selected, it contains two or three complete objects. The algorithm minimizes the sum square error between the selected patch and the selected target region, $d(o_to_{t+1}o_{t+2}, o_{c-1}o_co_{c+1})$. Periodic motion of the object is preserved in this process since the algorithm tries to find maximum correlation between visible parts of the partially occluded object in the hole and corresponding parts in the selected patch. The algorithm repeatedly selects all possible patches from the source region and matches them with the selected target region to find the final location of the best matching patch. For non-periodic object motions, a smooth transition between objects connecting left and right sections of the missing region is not guaranteed.

To improve the speed of the proposed algorithm, the priorities of the top and the bottom boundary points of the hole are not calculated. Fig. 4 shows the steps of foreground inpainting.

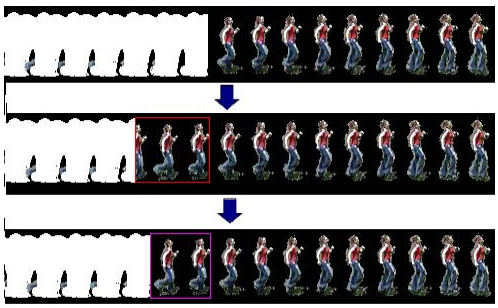

Fig. 4 Steps of foreground inpainting
The first row is the original mosaic image, the second row shows the primary selected patch, which is indicated by the rectangle, containing two complete objects and a section of another object, and the third row is the modified result after projection of the selected patch

Finally, the objects of each frame are isolated for constructing the video output. The frames of the video sequence containing the inpainted objects are obtained by placement of the objects in their locations. To improve the output, alpha matting can be applied.

3 Experiments and results

To compare our algorithm with other methods, three video sequences were used. Two of them had stationary backgrounds and the other had a non-stationary background. In the first video, the 'jumping girl', which was captured and used by Wexler *et al.* (2004), a girl moves from left to right and passes behind an occluding object (a person). Fig. 5 shows the results of the proposed algorithm on the first video.

Fig. 6 shows the comparison of the results of our algorithm with those of the space-time video completion (Wexler *et al.*, 2004). The corresponding results show that our algorithm represents the moving object better than Wexler *et al*. (2004)'s algorithm. Also, since our algorithm separates the moving foreground from the background, there is no over-smoothing in the background. Note that the motion of the object is periodic here, which helps to fill the large holes using other frames; otherwise the proposed algorithm might not perform well.

The second video was captured by a hand-held camera with a resolution of 540×432 pixels per frame. A mask for this video was created manually. Fig. 7 shows steps of the proposed algorithm for this video.

In the third video, which was used by Cheung *et al.* (2006) and Venkatesh *et al.* (2009), a man moves from right to left and crosses behind an occluding object (a board). In this video, the moving person

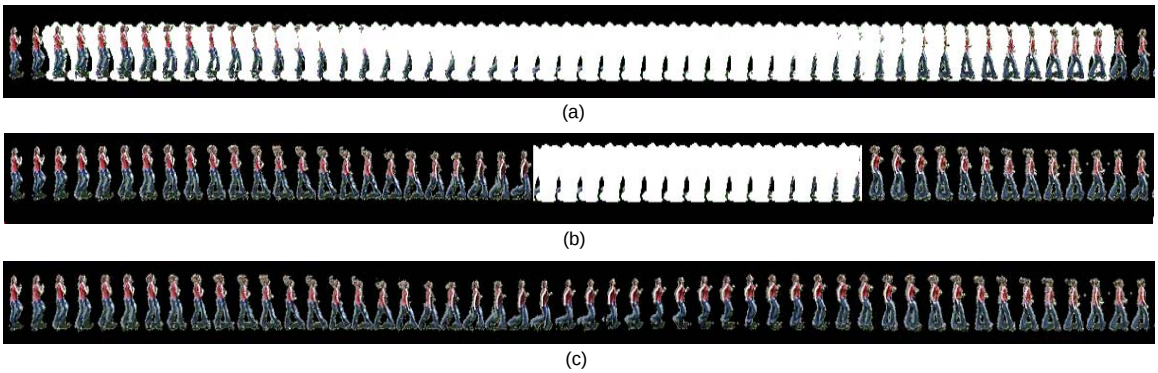

Fig. 5 Results of the proposed algorithm on the first video
(a) Original mosaic image with an occluded object; (b) Mosaic image after running several iterations of the algorithm; (c) Final result that has preserved continuity

Fig. 6 Comparison of the results of our algorithm with those of Wexler *et al*. (2004)'s algorithm on inpainting the jumping girl sequence
The first row shows the girl passing behind the occluding mask, the second row shows the inpainted sequence using our algorithm, and the third row shows the inpainted sequence using Wexler *et al*. (2004)'s algorithm

was completely occluded. Fig. 8 shows some frames of the original and the inpainted sequences using the proposed algorithm.

Fig. 9 shows results of our algorithm compared to results of an inpainting algorithm proposed by Cheung *et al.* (2006). As shown in the results of algorithm by Cheung *et al.* (2006), the direction of the object's movement changes in two consecutive frames when the object enters and exits from the occluding area. Results of our algorithm have smooth

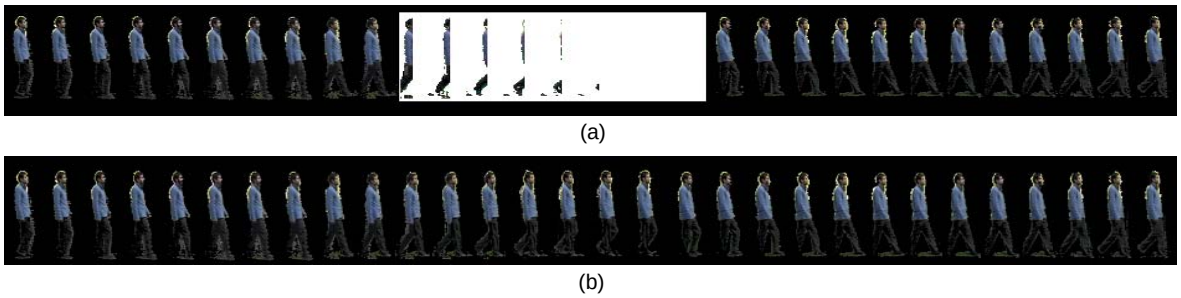

Fig. 7 Steps of the proposed algorithm for the second video
(a) Mosaic image with the occluded region; (b) Final result after applying our algorithm

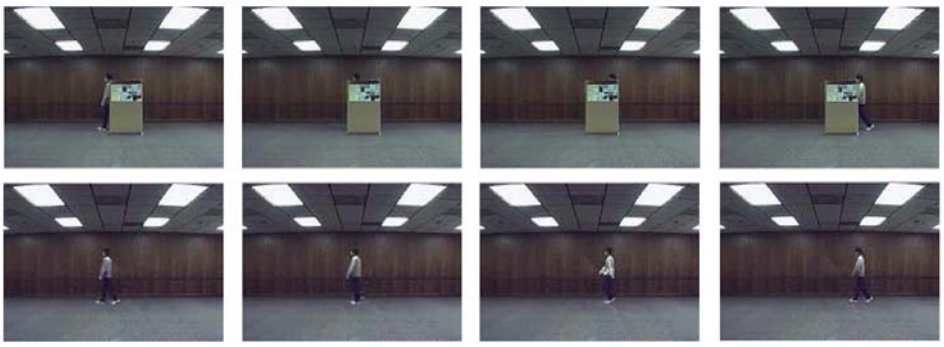

Fig. 8 Some frames of the original and the inpainted sequences using the proposed algorithm in inpainting an occluded object
The first row shows the input sequence with a board that occludes the object, and the second row shows the inpainted sequence using our algorithm

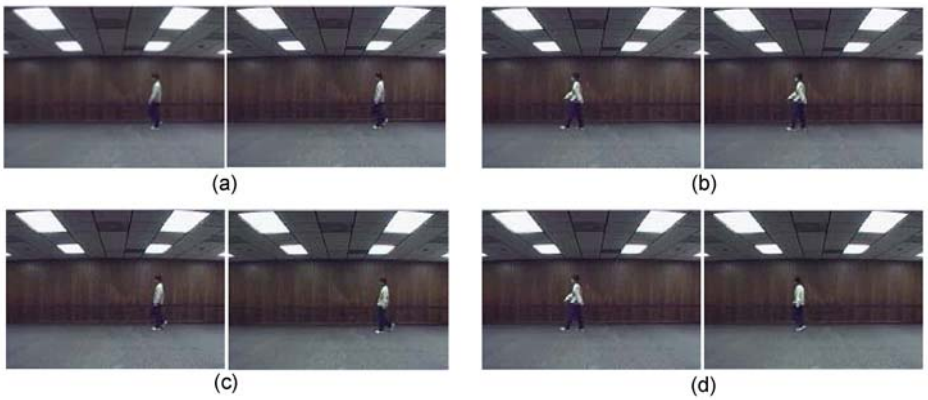

Fig. 9 Comparison of the results of our algorithm with those of an inpainting algorithm proposed by Cheung *et al.* (2006) in inpainting an occluded object
(a) and (b) show results of our algorithm when the object enters and exits from the occluding area, respectively; (c) and (d) show results of algorithm by Cheung *et al*. (2006) when the object enters and exits from the occluding area, respectively; (a) and (c) show two consecutive frames when the object enters the occluding area; (b) and (d) show two consecutive frames when the object exits from the occluding area. Note that the output of our algorithm has smooth transitions when the object enters and exits from the occluding area

transitions in the corresponding frames. Since the object's motion is non-periodic, there is one direction change in the final inpainted sequence using our proposed algorithm.

As mentioned above, background inpainting is done separately. Fig. 10 shows the results of the background inpainting for the first video sequence.

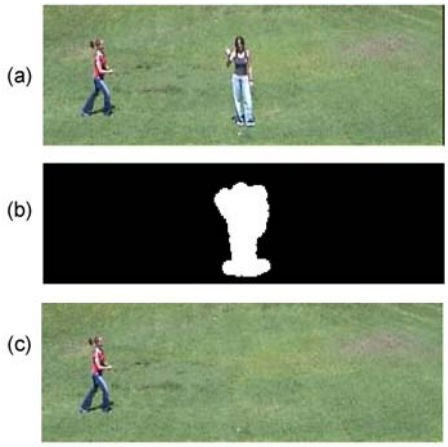

Fig. 10 Background inpainting of the first video
(a) Original frame with the occluding object; (b) Mask of frames; (c) Inpainted background

Results of background inpainting for the second and the third videos are shown in Figs. 11 and 12, respectively. Since the second video has a complex background, the inpainted background does not show an appropriate result.

After background completion, the final video is obtained by composing the inpainted foreground and background frames (the resulting video can be accessed at http://webpages.iust.ac.ir/koochari/proj/videoInpainting.html). To increase the speed of the proposed algorithm, priorities of the points on the top and the bottom boundaries of the target region have not been calculated.

4 Conclusion and future work

In this paper, we have implemented and tested a new method for video inpainting. It first separates the moving object from the background and then fills the missing data using a mosaic image and a patch-based image inpainting approach. The patches should be sufficiently large to cover the whole object. Finally, it places the objects in their foreground locations and

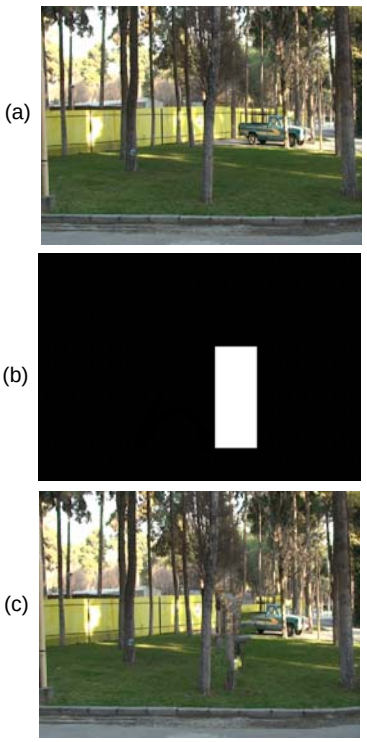

Fig. 11 Background inpainting of the second video
(a) Original frame; (b) Artificially created mask; (c) Inpainted background

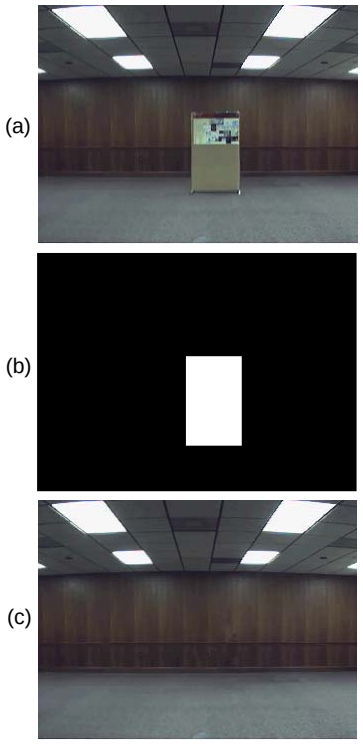

Fig. 12 Background inpainting of the third video
(a) Original frame with the occluding object; (b) Mask of frames; (c) Inpainted background

superimposes them with the inpainted background. The results of the algorithm are acceptable except for video sequences with non-stationary backgrounds. The main challenge in the proposed algorithm is to have smooth transitions between objects at connecting left and right sections of the mosaic image.

Future studies can be conducted in four areas: (1) representing objects in other domains such as a feature domain that can increase both quality and speed of the algorithm, (2) exploring other segmentation techniques for foreground/background separation, (3) developing algorithms that can maintain smooth transitions in damaged frames, and (4) considering more challenging situations such as moving camera and scale changes of the foreground object.

References

Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C., 2000. Image Inpainting. Proc. ACM SIGGRAPH Conf. on Computer Graphics, p.417-424. [doi:10.1145/344779.344972]

Bertalmio, M., Bertozzi, A.L., Sapiro, G., 2001. Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, **1**:355-362. [doi:10.1109/CVPR.2001.990497]

Bertalmio, M., Vese, L., Sapiro, G., Osher, S., 2003. Simultaneous structure and texture image inpainting. *IEEE Trans. Image Process.*, **12**(8):882-889. [doi:10.1109/TIP.2003.815261]

Cheung, S., Zhao, J., Venkatesh, M.V., 2006. Efficient Object-Based Video Inpainting. Proc. IEEE Int. Conf. on Image Processing, p.705-708. [doi:10.1109/ICIP.2006.312432]

Criminisi, A., Perez, P., Toyama, K., 2004. Region filling and object removal by exemplar-based inpainting. *IEEE Trans. Image Process.*, **13**(9):1200-1212. [doi:10.1109/TIP.2004.833105]

Ho, H.T., Goecke, R., 2007. Automatic Parametrisation for an Image Completion Method Based on Markov Random Fields. Proc. IEEE Int. Conf. on Image Processing, **3**:541-544. [doi:10.1109/ICIP.2007.4379366]

Liu, D., Sun, X., Wu, F., Li, S., Zhang, Y.Q., 2007. Image compression with edge-based inpainting. *IEEE Trans. Circ. Syst. Video Technol.*, **17**(10):1273-1287. [doi:10.1109/TCSVT.2007.903663]

Matsushita, Y., Ofek, E., Ge, W., Tang, X., Shum, H.Y., 2006. Full-frame video stabilization with motion inpainting. *IEEE Trans. Pattern Anal. Mach. Intell.*, **28**(7):1150-1163. [doi:10.1109/TPAMI.2006.141]

Oliveira, M.M., Bowen, B., McKenna, R., Chang, Y.S., 2001. Fast Digital Image Inpainting. Proc. Int. Conf. on Visualization, Imaging and Image Processing, p.261-266.

Patwardhan, K.A., Sapiro, G., Bertalmio, M., 2007. Video inpainting under constrained camera motion. *IEEE Trans. Image Process.*, **16**(2):545-553. [doi:10.1109/TIP.2006.888343]

Sheikh, Y., Shah, M., 2005. Bayesian Object Detection in Dynamic Scenes. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, **1**:74-79. [doi:10.1109/CVPR.2005.86]

Shen, Y., Lu, F., Cao, X., Foroosh, H., 2006. Video Completion for Perspective Camera under Constrained Motion. Proc. 18th Int. Conf. on Pattern Recognition, **3**:63-66. [doi:10.1109/ICPR.2006.1169]

Stauffer, C., Grimson, W.E.L., 1999. Adaptive Background Mixture Models for Real-Time Tracking. Proc. Computer Vision and Pattern Recognition, p.246-252. [doi:10.1109/CVPR.1999.784637]

Sun, J., Yuan, L., Jia, J., Shum, H.Y., 2005. Image completion with structure propagation. *ACM Trans. Graph.*, **24**(3):861-868. [doi:10.1145/1073204.1073274]

Venkatesh, M.V., Cheung, S.S., Zhao, J., 2009. Efficient object-based video inpainting. *Pattern Recogn. Lett.*, **30**(2):168-179. [doi:10.1016/j.patrec.2008.03.011]

Wang, H., Li, H., Li, B., 2007. Video Inpainting for Largely Occluded Moving Human. Proc. IEEE Int. Conf. on Multimedia and Expo, p.1719-1722. [doi:10.1109/ICME.2007.4285001]

Wexler, Y., Shechtman, E., Irani, M., 2004. Space-Time Video Completion. Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, **1**:120-127. [doi:10.1109/CVPR.2004.1315022]

Wexler, Y., Shechtman, E., Irani, M., 2007. Space-time completion of video. *IEEE Trans. Pattern Anal. Mach. Intell.*, **29**(3):463-476. [doi:10.1109/TPAMI.2007.60]

Wren, C.R., Azarbayejani, A., Darrell, T., Pentland, A.P., 1997. Pfinder: Real-time tracking of the human body. *IEEE Trans. Pattern Anal. Mach. Intell.*, **19**(7):780-785. [doi:10.1109/34.598236]

Zhang, Y., Xiao, J., Shah, M., 2005. Motion Layer Based Object Removal in Videos. Proc. IEEE Workshop on Applications of Computer Vision, p.516-521. [doi:10.1109/ACVMOT.2005.75]

Zivkovic, Z., 2004. Improved Adaptive Gaussian Mixture Model for Background Subtraction. Proc. 17th Int. Conf. on Pattern Recognition, **2**:28-31. [doi:10.1109/ICPR.2004.1333992]