2015 IEEE International Symposium on Mixed and Augmented Reality Workshops

Diminished Reality for Hiding a Pedestrian using Hand-held Camera

Kunihiro Hasegawa* Keio University

Hideo Saito† Keio University

ABSTRACT

This paper proposes a diminished reality method for hiding a pedestrian from a hand-held camera sequence. In the proposed method, the pedestrian is extracted from each frame of input video sequence using HOG based people detector. The detected area is excluded from each frame by masking, and then two frames are stitched for generating a stitched background image. By using the stitched background image, the pedestrian's mask for hiding is made. It is used for overlaying the background pixel to the pedestrian in the original frame. The presented experimental result is demonstrate the effectiveness of the proposed method.

Index Terms: H.5.1 [INFORMATION INTERFACES AND PRESENTATION]: Multimedia Information Systems—Artificial, augmented, and virtual realities; I.5.4 [IMAGE PROCESSING AND COMPUTER VISION]: Applications—Computer vision

1 INTRODUCTION

Augmented Reality (AR) and Mixed Reality (MR) are the technologies that present enhanced information on particular objects by overlaying virtual contents to real world images. There is diminished reality(DR) as a technology similar to AR. Diminished Reality (DR) is related technology to AR/MR, which visualizes hidden and unseen areas by replacing occluding objects with visual information of the hidden areas, instead of overlaying additional information onto the particular area.

A variety of DR studies have been performed. As typical examples, there are the methods utilizing large photo collections on the Internet[1], applying inpainting[2] and utilizing a depth sensor[3]. However, these methods have some limitations.

In this paper, we propose a method for synthesizing DR video in which moving objects from input video captured by a single hand-held camera. It can be applied to an outdoor scene since it does not need utilize a depth sensor or inpainting etc. Especially, we target to hide a pedestrian. One of the possible applications of such DR video synthesis is privacy protection. When we capture videos and photos of some outdoor scenes, there are some people walking around in the scene. By removing such pedestrian from the videos/photos, we can protect the privacy of those people. Therefore, it is desirable that a camera is available to be moved freely.

One of the related research to our method is DR using inpainting for video[4]. This method needs to make a mask for hiding area manually. By contrast, our method synthesizes background image by stitching from two frame images of an input video. In our method, a pedestrian's mask for hiding is made by using this background image automatically.

2 RELATED WORKS

Various studies are performed related to the DR or hiding objects. There are also a variety of the problem solving method. For example, there are methods using multiple cameras. Zokai et at. used cameras for capturing background image and project this image to main camera[5]. Although Zokai et al. used a static camera, Enomoto et at.[6] used hand-held cameras. Besides, Barnum et at. carried out to create an illusion of seeing moving objects through occluding surfaces[7]. Li et at. utilized large photo collections on the Internet instead of multiple cameras for achieving DR.

As a method without using multiple cameras, there are methods of pre-captured images and videos. Cosco et at. got the geometry of the object to hide preliminarily, then carried out DR using it[8]. In recent years, there is to obtain a three-dimensional geometric shapes by using a depth sensor[3].

As a method without using the information of direct real background image, there are the method utilizing inpainting. Inpainting is the method of reconstructing the missing area in the image from the surrounding information. As typical examples, there are Crisimisi's[9] and Wexler's[10][11] studies. Herling et at. utilized inpainting for hiding an object on simple texture[2]. Kawai et at. applied it to the scene a background is multiple planes[12].

Most related studies to our method are following studies. Granados et at. proposed a method for inpainting of region of dynamic object from a hand-held camera and by the framework of energy minimization[13]. Roxas et at. also proposed a method for inpainting from videos captured with a hand-held camera by estimating optical flow[4]. Those inpainting method need to make mask of the object regions by manual operation, while our method can synthesize DR video from hand-held camera by generating a mask automatically.

3 PROPOSED METHOD

3.1 Overview

In this study, we assume that the video is captured by a hand-held camera by a person who controls the camera to capture a pedestrian in each frame. It is a scene that the pedestrian walks straight in one direction. Other people are not in the background under this condition. In addition, the person controlling the camera in this scene stays at same place. He or she takes the video as catching the pedestrian at the center of the screen as much as possible. He or she has the camera in his or her hand and moves it freely to satisfy this condition.

Our method has two processing steps. Figure 1 is the flowchart of our proposed method. The first step is synthesizing a background stitched image and the second step is overlaying the background pixels to the pedestrian for hiding him or her. The background image is synthesized using images masked an area of pedestrian. For making this mask, we extract the pedestrian. This background image and input image make an other mask for overlaying. There are no restrictions about pedestrian's clothes and the background except that a color of the background and the clothes are not completely same.

3.2 Synthesis of Background Stitched Image

The background stitched image I_B is synthesized by a homography using feature points in each image. We use two images, a current frame image I_c and a past frame image before several frames I_p, for this synthesis. This is because to realize the process in real time. Stiching from two images is faster than that from many images. In a past frame, a pedestrian did not exist in the area where

*e-mail: hiro@hvrl.ics.keio.ac.jp
†e-mail:saito@hvrl.ics.keio.ac.jp

978-1-4673-8471-1/15 $31.00 © 2015 IEEE
DOI 10.1109/ISMARW.2015.18

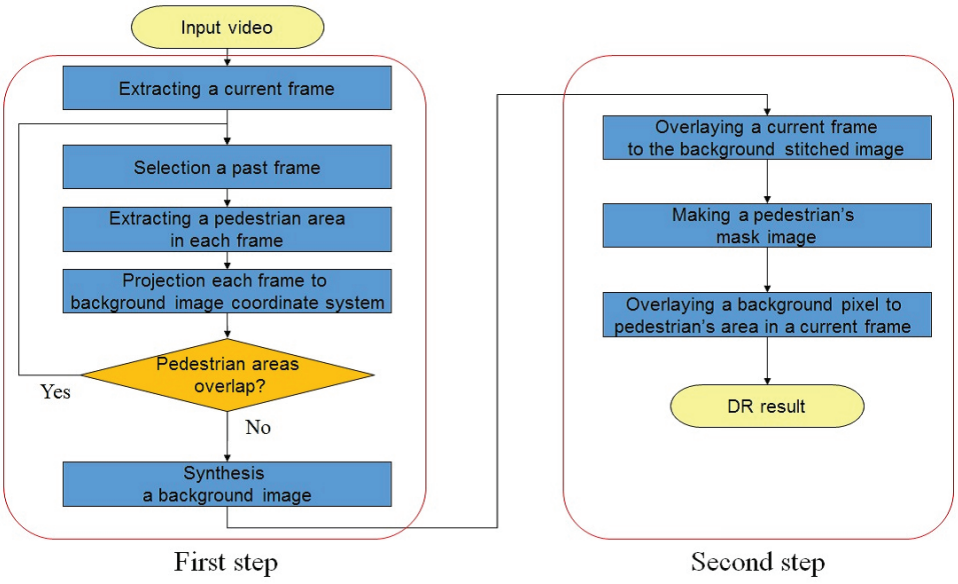


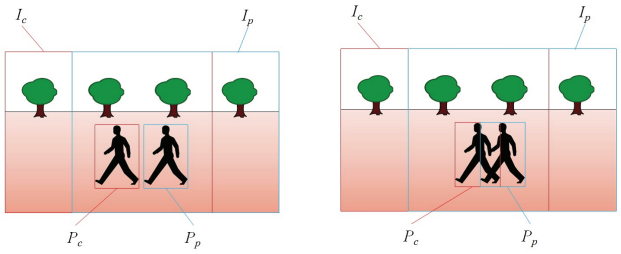
Figure 1: Flowchart of the proposed method

he or she exists in a current frame. Similarly, in a current frame, the pedestrian does not exist in the area where he or she existed in a past frame. Therefore, it is possible to synthesize the background stitched image if we combine these two images.

In those two frames, the bounding boxes should be separated as shown in Figure 2(a). If they are overlapped each other as shown in Figure 2(b), synthesized background image from those two images may have holes. To avoid this problem, we confirm these boxes do not overlap in the coordinate system of the background stitched image before stitching. Figure 2 is an example of a selection. The bounding box of pedestrian in a current frame P_c must not overlap to one in a past frame P_p. Figure 2(a) satisfy this condition. In this case, a synthesis of a background stitched image use these two frames. We select frames as close as possible from the combination of frames whose runner's bounding boxes are not overlapped. Since a scene does not change between these two frames, making a mask process described below can be carried out. On the other hand, Figure 2(b) has an overlapped area. The background stitched image is synthesized from these frames has a hole by overlapping. Therefore, we do not select this set of frames.

Searching for all past frames in every frame to find the frame satisfied this condition is too time consuming. Therefore, we forward the video sequence until the frame whose bounding box does not overlapped that of the first frame in the coordinate system of the background stitched image. We start DR processing from this frame. At this time, we save a difference of the number of frames between this frame and the first frame. Figure 3 is the flowchart of this processing. From next frame, the same processing is carried out using the previous frame by the saved number of frames. We assume that the pedestrian have a linear uniform motion. Of course, the pedestrian may change speed of walking. Especially, if the pedestrian slow down, overlapping may be occurred. Therefore, we process again after retargetting further previous frame if the overlapping occurs. We continue this processing until the last frame of a video sequence.

In order to extract feature points under the complex environment such as an outdoor, SURF[14] is used as feature points. The stitching method is based on the stitch function of OpenCV. We use a binary mask M_c for an image of current frame I_c instead of finding a seam mask for hiding the pedestrian in background stitched image without exception. The size of a binary mask M_c is same as that of I_c. The pixel value of M_c is 0 corresponding to the area that is inside of pedestrian's bounding box in I_c. The other area is 1. A past frame is same. Figure 4 is an example. Figure 4(b) is obtained by overlaying M_c to I_c. Figure 4(d) is also same. The mask area of each frame is ruled out in synthesizing the background image, hence only pixels of the background are used. HOG-SVM[15] is used for extracting a pedestrian. These methods can extract a human under the severe background, so enabling to make the mask in the condition of this study. Figure 5 is a result of synthesizing the background image from Figure 4(b) and Figure 4(d).



(a) Separate (b) Overlap

Figure 2: Diagrams of selected frames

3.3 Making a Mask for Overlaying

The second step makes the other binary mask O_c for overlaying. Figure 6 is an example of mask. It is used for overlaying a background pixel to a pedestrian in an original frame. We use this mask in order to leave the original image as much as possible. This mask is made by subtracting the background image from the original frame projected to the coordinate system of the background

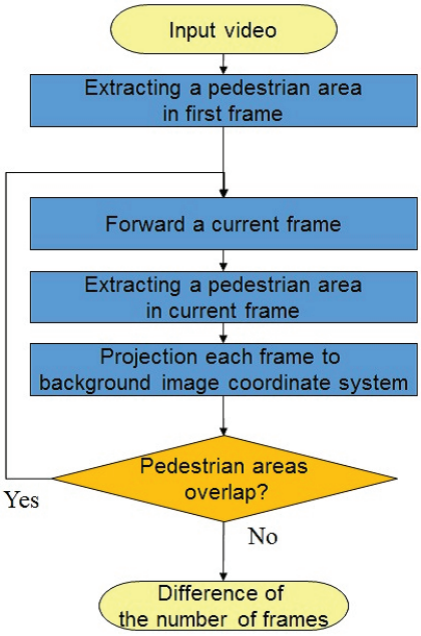
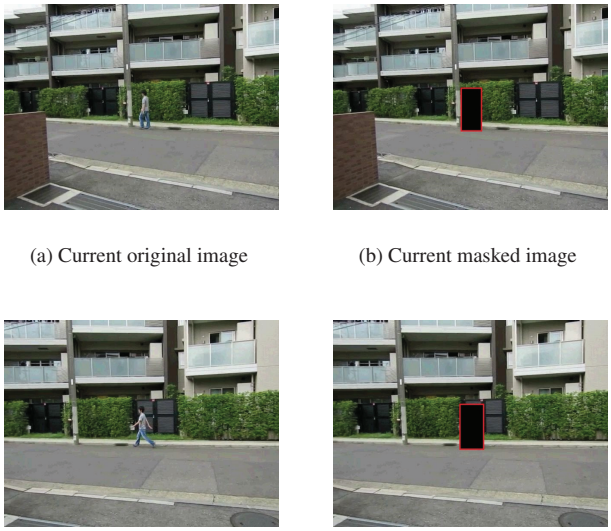


Figure 3: Flowchart of the first frame



(a) Current original image

(b) Current masked image

(c) Past original image

(d) Past masked image

Figure 4: Images for the background image synthesis

Figure 5: Synthesized background

stitched image. For example, Figure 4(a) and Figure 5 are used for making Figure 6. Figure 6 is obtained by subtraction of these two images and binarization. Only a part of pedestrian has a pixel value 1 such as Figure 7, ideally. In actually, it has a pixel value 0 on account of some reason, for example a part of pedestrian's color and that of background are similar. We carry out the morphology process to go away this problem, hence this is a minor deficit.

Only a pedestrian area in the background stitched image I_B is drawn in projected original frame by utilizing this mask O_c. The drawn frame is re-projected to the coordinate system of the original frame. Projection and re-projection utilize the homography for synthesis of the background image. Although the other region of the pedestrian is slightly appear, there is no problem since it does not have a major effect on the result. These series of processing makes the diminished reality result.

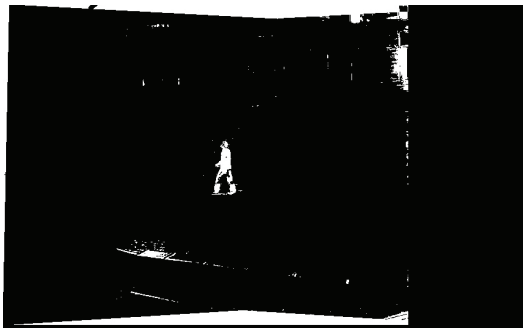


Figure 6: Example of a mask for hiding a pedestrian

4 Experiences and Results

The input video for all experiments was shot by a consumer digital video camera. As described in section3.1, a person held the video camera with own hand and moved the camera as tracking the pedestrian in shooting. We captured video images in four scenes in order to show that this method can be used in various environments. Figure 8(a), Figure 9(a), Figure 10(a) and Figure 11(a) show a part of the input frames used for the experiments. The size of image was 640 $\times$ 480 pixels and the number of frame was about 100. We used Microsoft Visual Studio 2010 as the IDE, C++ as the programming language and OpenCV 2.4.11 as the image processing library for implementation The specs of PC used to experiment were the CPU:Intel CORE i 7 2.40GHz / 4 cores, GPU:NVIDIA GeForce GTX 860M and the memories:8.00GB.

Figure 7: Ideally mask

Table 1: Processing time

	Ave(sec)	Max(sec)	Min(sec)
Scene 1	0.281	0.303	0.260
Scene 2	0.268	0.331	0.229
Scene 3	0.276	0.323	0.232
Scene 4	0.293	0.347	0.268

Figure 8(b), Figure 9(b), Figure 10(b) and Figure 11(b) are diminished reality results. Comparing each figure and input figure shows the pedestrian in each frame is hidden roughly. Table 1 shows average, maximum and minimum processing time for each frame in each scene. Every scene can process in about 0.28 seconds in average. In the other word, this is equal we can process in about 3.57fps in average. These results show that our proposed method can process almost in real time.

We notice a minor problem by looking at the images. In some images, the pedestrian can not be removed completely since the color of a hidden pedestrian area differs from around such as Figure 12. This is because a color may slightly changed between the past and the current frame.

Improving the computational speed is one of the issue of our research. Utilizing a camera tracking is one of the possible way to improve the computational speed, since we do not incorporate any information from temporally neighboring frames in the proposed method at present. Replacing the feature point detector with faster one, such as Orb[16] or Freak[17], is one of the way to improve the frame rate.

Proposed method is targeted for a scene that has only one pedestrian. However, we can apply it for more complex scenes, for example, plurality of pedestrians pass each other. Specifically, if you need to hide all pedestrians, we can apply this method by masking them simply. If the difference in speed between the pedestrian is large, it may be necessary to make a mask using a different frame for each pedestrian. On the other hand, if we need to hide only certain pedestrian, this method can not be used. We need to add another processing such as tracking each pedestrian while each pedestrian is identified as a different person. For example, we can remove all pedestrians by masking them by the proposed method. When we need to remove only selected few pedestrians from a lot of people, we need to apply some method for tracking the selected pedestrians.

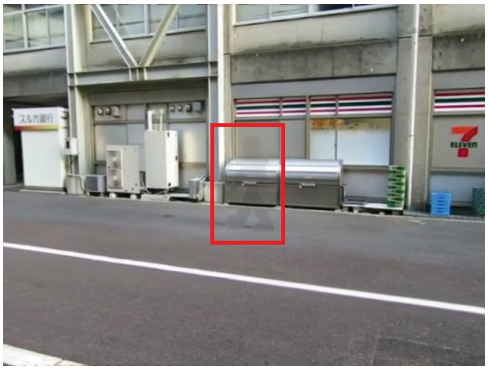


Figure 12: The color of a hidden pedestrian area differs from around

5 Conclusion

In this paper, we proposed a method of diminished reality for hiding a pedestrian using a hand-held camera. We carried out this goal by using masks. We confirmed the effectiveness of our proposed method by the results of the experiment. This study succeeded in diminished reality almost in real time. We will address further speed up and application to the environment has multiple people based on this proposed method.

Acknowledgements

This work was partially supported by JSPS Grant-in-Aid for Scientific Research(S) 24220004, and JST CREST "Intelligent Information Processing Systems Creating Co-Experience Knowledge and Wisdom with Human-Machine Harmonious Collaboration".

References

[1] Z. Li, Y. Wang, J. Guo, L-F. Cheong and S. Z. Zhou, "Diminished reality using appearance and 3D geometry of internet photo collections," IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp.11–19, 2013.

[2] J. Herling and W. Broll, "Pixmix: A real-time approach to high-quality diminished reality," IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp.141–150, 2012.

[3] H. Saito, T. Honda, Y. Nakayama and F. Sorbier, "Camera Pose Estimation for Mixed and Diminished Reality in FTV," 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), pp.1–4, 2014.

[4] M. Roxas, T. Shiratori and K. Ikeuchi, "Video Completion via Spatio-temporally Consistent Motion Inpainting," IPSJ Transactions on Computer Vision and Applications, Vol. 6, pp.98–102, 2014.

[5] S. Zokai, J. Esteve, Y. Genc, and N. Navab, "Multiview paraperspective projection model for diminished reality," IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp.217–226, 2003.

[6] A. Enomoto and H. Saito, "Diminished reality using multiple handheld cameras," Proceeding of Asian Conference on Computer Vision (ACCV)'07 Workshop on Multi-dimensional and Multi-view Image Processing, pp.130-135, 2007.

[7] P. Barnum, T. Sheikh, A. Datta, and T. Kanade, "Dynamic seethroughs: Synthesizing hidden views of moving objects," IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp.111–114, 2009.

[8] F. I. Cosco, C. Garre, F. Bruno, M. Muzzupappa and M. A. Otaduy, "Augmented touch without visual obtrusion," IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp.99–102, 2009.

[9] A. Criminisi, P. Perez, and K. Toyama, "Object removal by exemplar-based inpainting," IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2, pp..II–721–728), 2003.

(a) Input frames

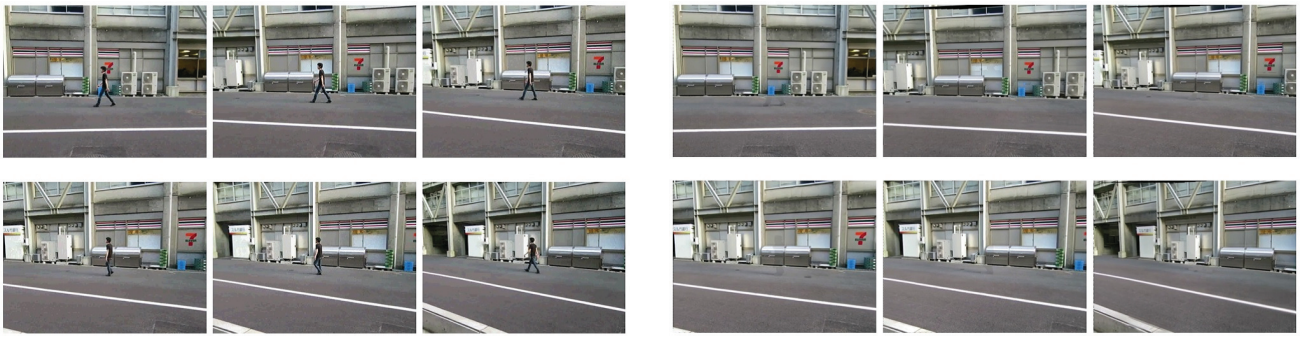
(b) Output frames

Figure 8: DR result in scene 1

(a) Input frames

(b) Output frames

Figure 9: DR result in scene 2



(a) Input frames

(b) Output frames

Figure 10: DR result in scene 3

(a) Input frames

(b) Output frames

Figure 11: DR result in scene 4

[10] Y. Wexler, E. Shechtman,and M. Irani, “Space-time completion of video,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.29, no.3, pp.463–476, 2007.

[11] Y. Wexler, E. Shechtman,and M. Irani, “Space-time video completion,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Vol.1, pp.I–120–127), 2004.

[12] N. Kawai, T. Sato and N. Yokoya, “Diminished reality considering background structures,” IEEE International Symposium Mixed and Augmented Reality (ISMAR), pp.259–260, 2013.

[13] M. Granados, K. I. Kim, J. Tompkin, J. Kautz and C. Theobalt, “Background Inpainting for Videos with Dynamic Objects and a Free-Moving Camera,” European Conference on Computer Vision(ECCV), pp.682–695, 2012.

[14] H. Bay, A. Ess, T. Tuytelaars and L. V. Gool, “SURF: Speeded Up Robust Features,” Computer Vision and Image Understanding, vol.110, no.3, pp.346–359, 2008.

[15] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), pp.886–893, 2005.

[16] E. Rublee, V.Rabaud, K. Konolige and G. Bradski, “ORB: an efficient alternative to SIFT or SURF,” IEEE International Conference on Computer Vision (ICCV), pp.2564-2571, 2011.

[17] A. Alahi, R. Ortiz, and P. Vandergheynst, “Freak: Fast retina keypoint,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 510–517, 2012.