Video Inpainting for Object Removal and Video Restoration

Rong-Chi Chang
Department of Media & Design, Asia University, Taichung, 413, Taiwan, R.O.C.
roger@asia.edu.tw

Nick C. Tang
Dept. of Computer Science & Information Engineering, Tamkang University, Taiwan, R.O.C.
nick@mail.mine.tku.edu.tw

ABSTRACT

Video inpainting is the technique that can automatically restore damaged or partially removed image. It is also the unique tool for filling in the missing part in a video sequence. Exploration of more advanced concepts in video painting, we develop a new video algorithm based on temporal continuations and exemplar-based image inpainting techniques. This proposed algorithm involves the elements of removing the moving objects on stationary and non-stationary background. Therefore, the related experiments include a wide variety of temporal continuations of foreground and background. According to the results of our experiments, a motion-compensated inpainting procedure is successfully developed and it can be further extended to implant into the inpainted background video. Ultimately, the inpainted video backgrounds are visually pleasant with smooth transitions.

Keywords: Video inpainting, Object removal, Motion estimation, Image completion, Digital, inpainting

1. INTRODUCTION

Digital inpainting is an interesting new research topic in multimedia computing and image processing since 2000 [1, 2, 4]. In the literature, the first intention of image inpainting was to remove damages portions of an aged photo, by completing the area with surrounding or global information. The techniques used include the analysis and usage of pixel properties in spatial and frequency domains. Furthermore, image inpainting techniques were used in object removal (or image completion) in photos. Several strategic algorithms were developed based on confidence values and priorities of using pixels. The techniques used in still images were then extended to video inpainting, which need to consider temporal properties such as motion vectors. With a reasonable combination of object tracking and image completion, objects in video can be removed and possibly replaced. On the other hand, aged films contain two types of defects: spikes and lone vertical lines. These defects need to be precisely detected and removed to restore the original film. In addition, based on image completion techniques, incompleteness of scanning results of a 3-D scanner due to improper location or other reasons of a scanner can be solved.

Unlike image inpainting that user can select a target area to be inpainted, the problem in video inpainting is to detect damages, such as spikes or lone scratch lines. Detection of defects in video frames is a must for motion picture restoration. Defects in video frames include spikes (usually in a bright or dark intensity) [3] and long vertical lines (usually in a dark intensity and in a large length) [4-6]. The former is due to dust and dirt, which occurs randomly and mostly appears in one or two frames. The later is accidentally produced in film development and usually occurs in the same position for an unknown duration (from several frames to several minutes). Detections of the two types of damages are different problems. However, both problems need to look at the damages from temporal and spatial domains. Once the defects are detected, image inpainting technique can be used to repair spikes or lines approximately. However, it is also possible to use realistic data in different frames to achieve a better repairing result. Thus, motion estimation can be used to find suitable blocks among continuous frames. In addition to repairing defects, video inpainting can be applied to logo removal [7] and object removal [9]. Logos could be annoying. Usually, they occur in the same position. Thus, to detect the position of logos is not as hard as tracing spikes or lines. Once the logo position is decided, image inpainting technique is used in [7] to remove the logos in all frames. Removing objects or part of objects from a stationary background was presented in [8]. An optical-flow based mask is decided based on moving pixels. Then, the moving object is inpainted using information in undamaged frames. Calculation of confidence values of source pixels is used to ensure good inpainting results. In addition, a motion layer extraction mechanism discussed in [10] is able to separate a video sequence into different layers based on overlapping order. A layer with a target object is then selected. Finally, the targeting object is removed using motion compensation and region completion algorithms. These works discussed in [8] and [10] remove objects from a stationary or dynamic background.

The purpose of this study is to develop a motion-compensated inpainting procedure. In order to produce the authentic algorithm, varieties elements of temporal continuations of foreground and background are included in the experiments. Moreover, a simple tracking algorithm

is used to conduct a moving object and this object will be implanted into the inpainted background video. Finally, the rest of this paper is organized as follows: The algorithms are described in Section 2 followed by experimental results in Section 3 and then the conclusions as well as future work are provided in Section 4.

2. THE PROPOSED VIDEO INPAINTING METHOD

Video clips can be generated by computer or taken by video camera. We consider both stationary and non-stationary videos since their temporal continuations of background are different. This section presents the proposed method. We first give an overview of the algorithm in terms of its computational steps, and then discuss the steps in detail. We make the following assumptions about the video to be processed: the camera is fixed; the scene is composed of stationary background with a moving object (i.e., humans or car); the moving object to be repaired has periodic motion.

Our approach consists of the following major steps:

(1) Background Construction Method:

The background information is constructed based on the video sequence and certain reference links to the rough information on moving objects. We tend to integrate the existing approaches with the improved algorithm for this task.

(2) Object Region Decision:

The focus of this step is to determine the object region by finding the frame differences in video sequence.

(3)Moving-Object Detection:

Considering different types of video, certain features of video sequence are included in this step, including stationary background with moving objects and non-stationary background with slow foreground. Therefore, using the tracking algorithm becomes essential when the motion estimation is conducted to decrease the tracking area.

(4) Video Inpainting on Object Removed:

The modified image inpainting algorithm is applied to all frames in a video clip. Objects to be removed are tracked by a tracking algorithm and the related issues are discussed in subsection 2.4. Each inpainted frame has a visually pleasant result when it is viewed individually.

2.1 Background Construction Method

The basic idea of this rough and reliable background construction algorithm is bases on change detection. In paper [10], the change detection approach separated the difference frame into the changed region and the unchanged region, according to a threshold obtained from background estimation. The difference between two consecutive input frames is the basic concept of change detection. Rough and consistent object information is very difficult to obtain because the behavior and characteristics of the moving objects differ significantly. The quality of segmentation result depends strongly on the background noise, object motion, and the contrast between the object and the background. While the algorithm, which are based on inter-frame change detection, enable automatic detection of objects and allow larger non-grid motion compared to object tracking methods and object boundaries tend to be irregular in some critical image areas due to the lack of spatial edge information. Instead of trying to obtain more information from the moving objects of the video sequence, the focus is on the rough background. The long-term behavior of the object motion accumulated from the several frames instead of relying on frame difference of two consecutive frames that causes the final result more rough. The block diagram of rough background construction is display in Figure 1. The first step is to calculate the frame difference mask by thresholding frame difference that comes from two consecutive frames and ten frame difference masks are generated. The information of ten frame difference masks is counted in the background buffer. According the frame difference masks of past several frames, pixels that are not moving for a long time are considered as rough background.

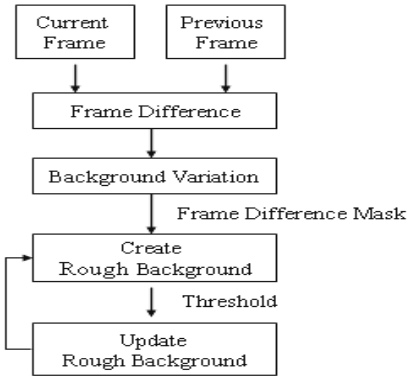


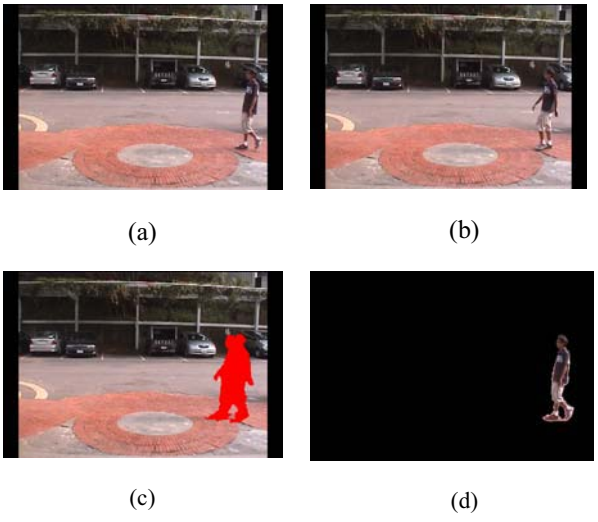
Figure 1. The block diagram of the rough background construction

2.2 Object Region Decision

We check the difference of mark in the video sequence and the difference of image mask of the background to decide the object region. Table 1 lists the criteria for distinguishing the regions, where "No" and "Yes" mean the pixel on frame difference (or background subtraction) mask to be the unchanged and changed, respectively. Therefore, the initial object mask can be obtained, which combines the still region mask and the moving region mask. The sample result of object region decision shown in Figure 2.

Table 1. Object region detection rule

Region type	Background subtraction mask	Frame difference mask
Background	No	No
Uncovered background	No	Yes
Still region	Yes	No
Moving region	Yes	Yes



(a) (b)

(c) (d)

Figure 2. Illustration of object region decision: (a) and (b) are the frames from original video sequence. (c) demonstrates the frame difference by showing red color between two consecutive input frames. (d) presents the result of the object region decision.

2.3 Moving-Object Detection

In our video inpainting algorithm, it is necessary to determine whether there is any moving object contained in the input image sequence. Thus, an image frame will be viewed as to be composed of foreground (i.e., moving object) and background as shown in Figure 3.

Figure 3. Background and foreground

(A) Moving Object Detection on Stationary Video

Firstly, it is necessary to determine whether there is any moving object contained in the input image sequence. Thus, an image frame will be viewed as to be composed of background and foreground (i.e., moving object) as shown in Figure 3. Because of using change detection approach for segmenting that moving object, it requires to obtain the absolute background for reference. By averaging several continuous frames, the rough background, denoted by $BG(x, y)$, is derived in the following equation.

$$BG(x,y) = \frac{\sum_{t=1}^{t=n} frame(x,y,t)}{n} \qquad (1)$$

where $frame(x, y, t)$ is the image frame at time t and n is the selected number of frames for estimating the rough background. The above rough background will be updated when the frame meets scene change, but it will be not updated once the moving object appears in the frame.

Based on equation (2), the target area mask $TAM(x, y, t)$ of moving regions is obtained for further moving-object extraction.

$$TAM(x,y,t) = \begin{cases} 255, & \text{if } \left| frame_I(x,y,t) - BG_I(x,y) \right| > W \cdot D_T \\ 0, & \text{if } \left| frame_I(x,y,t) - BG_I(x,y) \right| \le W \cdot D_T \end{cases} \qquad (2)$$

where the subscript I (i.e., intensity) denotes the illumination component (i.e., gray-level plane) for their frames, and W and D_T is a weighted value and a temporal standard deviation, respectively, and their product is employed for motion judging. For a pixel, if the absolute value of the difference between the current frame and rough background frame in gray-level is greater than $W \cdot D_T$, the pixel is judged as a moving pixel and thus the value of the corresponding position in the binary mask is set to 255; otherwise, it is set to zero. Then, equation (3) is used to judge whether a region composed of those of pixels of 255 in the target area mask is a moving object or not.

If $M_O > T_{MO}$ **then** moving object

else not moving object (3)

where M_O means a region of moving object with value 255 in the binary mask and T_{MO} is a threshold of area of a moving object. Finally, that binary mask needs to be modified through morphological processing for removing spot noises.

(B) Moving Object Detection on Non-Stationary Video

Tracking on non-station video is more complicated. Yet, a few techniques can be used from the literature. For instance, tracking based on optical flow estimates the motions of foreground objects. Yet, the technique needs to separate video into several layers before each moving object can be identified. Optical flow can be computed by using estimation of motion vectors of blocks (e.g., 8 by 8 pixels as a block). Typical estimation computes the minimum difference between target block and its corresponding estimated block. In the literature, MAD (Mean Absolute Difference), SSD (Sum of Square Difference), and SAD (Sum of Absolute Differences) are used to estimate the differences caused by movements of blocks. Assuming that the block size is p by q, these formulae can be defined as the following:

$$MAD_{(i,j)}(dx,dy) = \frac{1}{p \times q} \sum_{a=0}^{p-1} \sum_{b=0}^{q-1} \left| I_n(i+a, j+b) - I_{n+1}(i+a+dx, j+b+dy) \right| \quad (4)$$

$$SSD_{(i,j)}(dx,dy) = \sum_{a=0}^{p-1} \sum_{b=0}^{q-1} W(i,j) \left| I_n(i+a, i+b) - I_{n+1}(i+a+dx, j+b+dy) \right|^2 \quad (5)$$

$$SAD_{(i,j)}(dx,dy) = \sum_{a=0}^{p-1} \sum_{b=0}^{q-1} \left| I_n(i+a, j+b) - I_{n+1}(i+a+dx, j+b+dy) \right| \quad (6)$$

where (i, j) is a reference point of a block, and (dx, dy) represents the offset of the movement. The intensity I_n is compared to the intensity I_{n+1}. A motion map consists of estimated minimal vectors can be obtained. The following figure is a simple tool developed in our lab to estimate the motion vectors. Yet, the motion map needs to be further constrained to reduce noises, by using morphological operators.

The above tool illustrates a very simple mechanism on stationary video. The RGB color space is used. However, similar technique and other color spaces can be used to enhance the results. In fact, for non-stationary videos, advanced techniques are required (will be investigated with the analysis of temporal continuations).

Figure 4. The motion vectors estimation tool

2.4 Video Inpainting on Object Removed

Our discussion strictly follows the notations introduced in [9]. Let $\boldsymbol{I}$ be the original image (or a frame in a video) which includes a target area, denoted by Ω, to be inpainted and a source area, denoted by Φ, where patches are searched and used. Hence, $\boldsymbol{I} = \Phi \cup \Omega$. A simple region segmentation algorithm based on the CIE Lab color space is used to convert $\boldsymbol{I}$ to $\boldsymbol{I'}$. Let p_i and p_j be pixels and s_i and s_j be segments.

$\forall p_i \in I$, $\forall p_j \in I$, $p_i \neq p_j$ and p_i is adjacent to p_j,

$SSD_{CIE\ Lab}(p_i, p_j) < \delta_c \Rightarrow$ make p_i, p_j in the same segment;

$\forall s_i \in I$, $\forall s_j \in I$, s_i is adjacent to s_j,

$p_n(s_i) - p_n(s_j) < \delta p_n \Rightarrow$ make s_i, s_j in the same segment,

where $p_n(s)$ computes the number of pixels in a segment. And, $SSD_{CIE\ Lab}(p_i, p_j)$ calculates the SSDs using the CIE Lab color space.

The algorithm for video inpainting is very complicated. In general, it can be described as the following:

Step 1. The target area Ω^1 in the first frame is manually selected and tracked as Ω^t through the entire video. Bounding boxes $\underline{\Omega}^t$ are computed.

Step 2. Inpaint Ω^1 using the image inpainting algorithm.

Step 3. For each Ω^t, $2 \leqq t \leqq$ last-frame,

Step 3-1. Compute the difference, $\mu^t = \Omega^{t-1} - \Omega^t$.

Step 3-2. Compensate and copy patches in μ^t by $\Delta_{x,y}$ (and $\chi_{x,y}$) from frame t-1 to frame t.

Step 3-3. Inpaint $\Omega^t \setminus \mu^t$ using the image inpainting algorithm

We use several parameters which are summarized below with best values:

- Color distance for segmentation: $\delta_c = 3$
- Pixel number of groups for color segmentation: δ_{pn} is between 50 and 100, depends on video
- Size of patch: $|\Psi_p| = 5 \times 5$
- Distance to search: $r = 15$
- Neighbor distance index for patch template: k=1

The first two parameters influence the edge map Φε. The use of δ_{pn} for video games is smaller as compared to scenery videos in general. The size of patches is set to 5 by 5 for all videos. A patch size smaller will make the search less accurate thus the confidence term fails to prioritize patch candidates. A larger patch size results in a less realistic inpainting result. We also found that the distance for searching patches does not significantly affect inpainting results. Thus, we use r = 15. The index for patch template is set to 1. Thus, a patch template is 9 times of its patch in size. On the other hand, several interesting observations were found due to temporal inpainting. Searching for patch templates among different frames does not increase the visual quality of image inpainting on a particular frame, since we only use short video scenes which contain similar background in all frames.

3. EXPERIMENTS RESULTS

We use a simple tracking technique to locate moving objects in static and non-static background of video clips. The objects are removed from frames. The temporal and spatial inpainting algorithms we have developed for the above two types of restoration are used to produce two short sequences in Figures 5-7. In the first experiment (Figure 5), we have successfully removed the sportsman in the center. Observe that the inpainted background is consistent throughout the video. In Figure 6, we shown the foreground object (car) have be removed on the stationary background. Figure 7 shown the moving person have successfully removed on a non-stationary background video. Our algorithm performs very well even when the region to be inpainted is very large.

4. CONCLUSIONS

Employing the video inpainting technique under a full set of camera motions is a challenging task. In this paper, we propose a novel algorithm that integrates with several inpainting techniques, including background replacement, image inpainting and object interpolation for video inpainting techniques. One of our major contributions in this study is to develop a motion-compensated video inpainting procedure based on the novel inpainting algorithm and the computation of motion stabilizer vectors. We conclude that different temporal continuations of foreground and background should be treated differently with different inpainting strategies. Experimental results prove that our proposed algorithm can produce visually pleasant results.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Council of the Republic of China under contract NSC 94-2218-E-468-002 and NSC 95-2221-E-468-007.

Figure 5. Object removal and inpainting with stationary background. Top row: Five selected frames from the original car sequence. Bottom row: The correspondent frames obtained by our algorithm, in which the sportsman was removed.

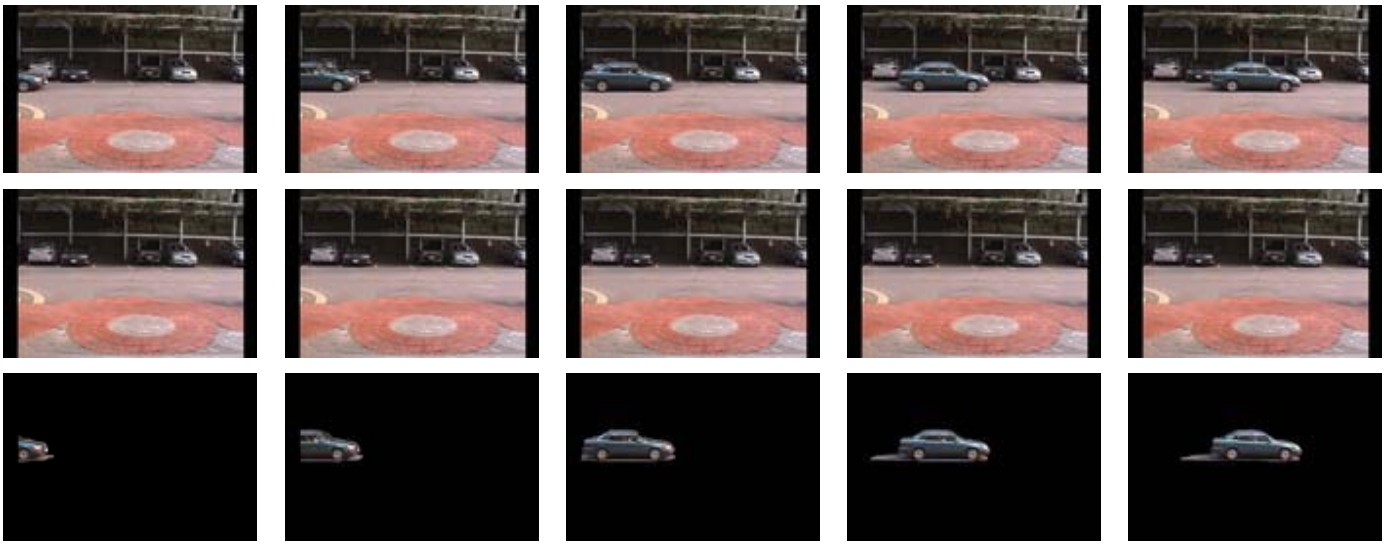


Figure 6. Video inpainting with stationary background. Top row: Five selected frames from the original car sequence. Middle row: The correspondent frames obtained by our algorithm, in which the car was removed. Bottom row: The object (car) extracted form the original frame.

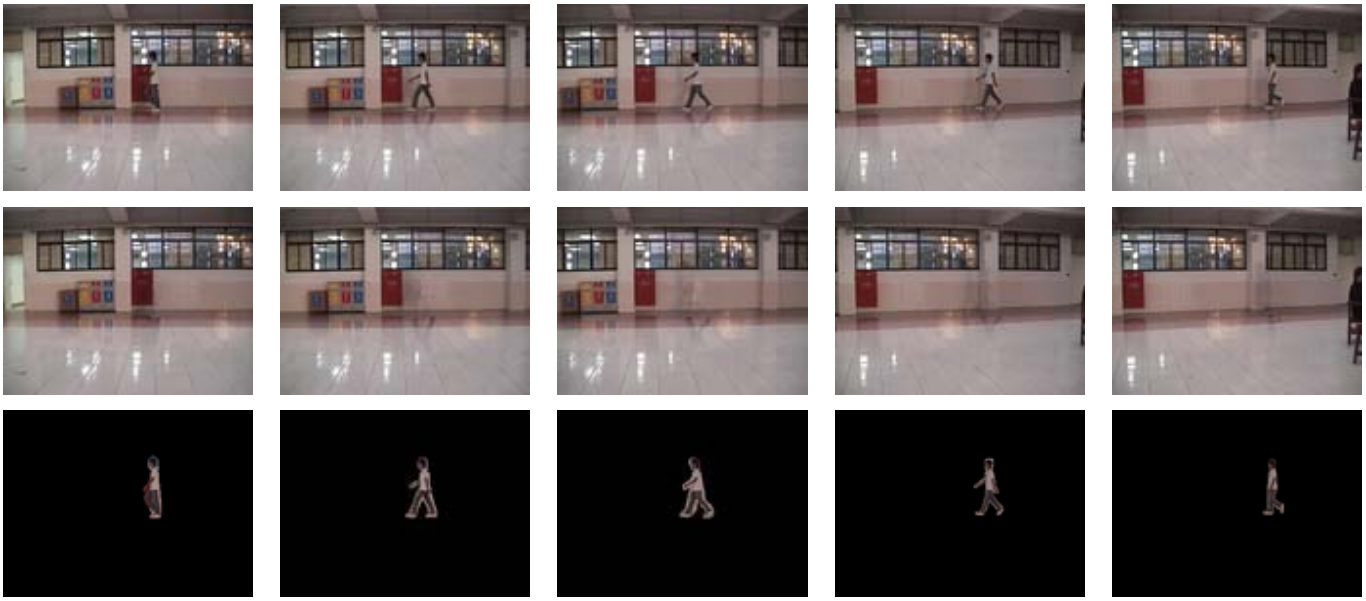


Figure 7. Video inpainting with non-stationary background. Top row: Five selected frames from the original video sequence. Middle row: The correspondent frames obtained by our algorithm, in which the human was removed. Bottom row: The human extracted form the original frame.

REFERENCES

[1] Bertalmio, M., Sapiro, G., Caselles V., and Ballester C., "Image Inpainting", in Proceedings of the ACM SIGGRAPH Conference on Computer Graphics, 2000, pp.417-424.

[2] Bertalmio, M., Vese, L., Sapiro, G., Osher, S.,"Simultaneous structure and texture image inpainting", IEEE Transactions on Image Processing, vol. 12, no. 8 , 2003, pp. 882 – 889.

[3] Machi, A., Collura, F., "Accurate Spatio-Temporal Restoration of Compact Single Frame Defects in Aged Motion Picture", The 12th International Conference on Image Analysis and Processing, 2003, pp. 454 – 459.

[4] Joyeux, L., Buisson, O., Besserer, B., Boukir, S. "Detection and removal of line scratches in motion picture films", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1 ,1999, pp. 549 – 553

[5] Boukir, S.; Suter, D., "Application of rigid motion geometry to film restoration", The 16th International Conference on Pattern Recognition Proceedings ,vol. 1, 2002, pp.360 – 363

[6] Bruni, V.; Vitulano, D., "A generalized model for scratch detection", IEEE Transactions on Image Processing, vol. 13, 2004, pp.44 – 50

[7] Yamauchi, H.; Haber, J.; Seidel, H.-P."Image restoration using multiresolution texture synthesis and image inpainting", Computer Graphics International, 2003, pp.108 – 113

[8] Kedar A. Patwardhan, Guillermo Sapiro, and Marcelo Bertalmio, "Video Inpainting of Occluding and Occluded Objects", The 2005 IEEE International Conference on Image Processing, 2005, pp. II-69 – 72

[9] Bartesaghi, A., Sapiro, G., "Tracking of moving objects under severe and total occlusions", IEEE International Conference on Image Processing, vol. 1, 2005, pp. I-301 – 304

[10] Zhang, Yunjun; Xiao, Jiangjian; and Shah, Mubarak; "Motion Layer Based Object Removal in Videos", The 7th IEEE Workshops on Application of Computer Vision, 2005, pp. 516 – 521

[11] Criminisi, A., Perez, P., and Toyama K., "Object Removal by Exemplar-Based Inpainting", IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, 2003, pp. 721 – 728

[12] Chen, T.-H., Chen, T.-Y. and Chiou, Y.-C., "An Efficient Real-time Video Object Segmentation Algorithm Based on Chang Detection and Background Updating", IEEE International Conference on Image Processing, 2006, pp. 1837 – 1840