Contour-Based Video Inpainting

Amanna Ghanbari
School of Computer Engineering
Iran University of Science and Technology
Tehran, Iran
a_ghanbari@comp.iust.ac.ir

Mohsen Soryani
School of Computer Engineering
Iran University of Science and Technology
Tehran, Iran
soryani@iust.ac.ir

Abstract— **Inpainting is the technique that restores corrupted or partially damaged images and video frames. This paper proposes an algorithm for video inpainting when an object is totally damaged. In this framework the background and the moving object are separated from each other. By using these separated moving objects, a large mosaic image is constructed. Then a patch-based method with the help of a contour-based method and large patches fills the holes. By settling the objects on their places in each frame, the inpainted foreground is acquired. Missing regions of the stationary background are filled separately. Superimposing the inpainted foreground and the stationary background produces the inpainted video.**

Keywords-Video inpainting; contour; patch; background modelling; image completion

I. INTRODUCTION

Image and video inpainting are two interesting concepts in image processing, which are used to reconstruct damaged portions of pictures and video sequences. The earliest intention of image inpainting was to remove those parts of an aged photo that were damaged. But later some video inpainting methods were introduced to handle a wide range of applications such as video restoration and post processing [1]. Three examples of such applications are [2]: 1) To remove objects from video; typically, video frames have some static and dynamic objects which we would rather not to keep them in the video. Using video inpainting, we can delete those objects from video. 2) Old video restoration; old videos often have so much noise and are damaged visually. Some times parts of a video are lost or there is severe noise in them. These problems can mostly be solved by video inpainting. 3) Video story modifying; video inpainting can change people’s action who are on the scene. Video data from other parts of the video can help us with this application. Although this application may introduce some sociological problems, but if we use this technique in a good manner, we can have some interesting applications such as special effects of movies.

Image inpainting methods do not naturally extend to consider time [3]. A direct extension of image inpainting methods to video inpainting means we consider the video data as a set of images distinct from each other. Then we would use image inpainting methods to these distinct images individually. This approach is not useful for video inpainting because it does not take advantage of the high temporal correlation between video frames [4].

In video sequences with dynamic scenes temporal aliasing is essentially more important than spatial aliasing.

Temporal continuity of object’s moving must be preserved in a video sequence, so none of image inpainting algorithms will directly be used for video inpainting. The first attempt in this area was the one which has been introduced in [5]. It uses Partial Differential Equations (PDE) for all video frames to preserve continuity. Texture has not been considered here and the parameters were not set automatically. [6] combined motion vectors and image inpainting to complete the video. This method has a pre-processing step which separates moving foreground from the background. This method uses information of moving object from other frames as much as it can to fill missing data. Other parts can be inpainted using the local background. The overall performance of this method is good, but it does not work well when the missing region is large and the moving object is near to this region or enters it. Also cases such as camera movement (zooming), an entirely missing object and dynamic backgrounds were not considered in this paper. [3] used an objective function to complete the video as a global optimization problem. Similarity measure here was a three dimensional cuboids with a fixed scale. To estimate the color of the missed pixel, more than one of these cuboids are used; so, running time is the bottleneck of this approach. With pruning the dataset we can overcome this problem. [7] combined patch-matching with exemplar-based image inpainting to inpaint video. Applying motion segmentation and a modified procedure for video inpainting solved the problem of ghost shadows, partially. [8] finds a set of descriptors which encapsulate the information that is necessary to restore a frame. For applications such as inpainting, mapping the available pixels into a space with lower dimensions can obtain the descriptors. Once these descriptors are obtained, we can find an optimal estimation which is used to reconstruct the frames. The basic disadvantage of this method is not considering the changing of the scale or shape of the moving object. [9] removes moving object in both a static background and a dynamic background. This approach needs four basic steps: 1) Reconstruct background based on video sequence and information about the moving object. 2) Object region is found based on frame differences. 3) Moving object can be recognized based on whether the video has a static background and a moving object or it has a dynamic background and a slowly moving foreground. 4) Using tracking algorithm, the moving object will be tracked and the missing region will be inpainted. [10] considered more

978-1-4577-1535-8/11/$26.00 ©2011 IEEE

complex camera motions than motions which was considered in [8]. It also enhanced results of that method but still much more complex camera movements can be considered. [11] proposed a new mechanism for patch searching and a new strategy for patch adjustment to preserve the temporal continuity of video. Because of this, the results of video inpainting will improve. The drawback of this approach is sensitivity to the quality of the original video. In the case of a very bad quality video, this approach fails to reconstruct all of the defects. [12] constructs mosaics of background based on camera motion estimation and uses these mosaics to complete the background. This approach can also be used to complete dynamic textured background, because of using a model based on the linear dynamic system. In this case we may have a fixed camera with a dynamic background or a moving camera with a still background. [13] separates the moving object from the background. Using the moving object, a large mosaic is constructed. After that, a patch-based method fills the holes by using large patches. This framework considers neither camera motion nor foreground objects' scale changing. Besides, this approach considers a periodic motion for the foreground object.

Recent attempts on the field of video inpainting use a two-step procedure. They usually separate the video to two individual parts; foreground and background. Then completion is done to any of them, individually.

As described before, in video inpainting, the goal is to remove the unwanted objects and replace them with naturally or synthesized information in some way not to be visually characterized from the background [12]. Human's eye is sensitive to motion and this sensitivity is much stronger than eye's sensitivity to the color. Because of this, temporal consistency is more important than spatial consistency in video completion [3]. In video inpainting, to check the result, preserving the visual quality of the video should be considered. So, it is important to have a natural video inpainting result which is pleasant to the viewer. To have this, spatial and time continuity should be preserved [12].

This paper proposes a framework to fill damaged portions of video frames. There is an occluding object that causes missing data. The assumption is to have a stationary background and a periodic motion for the foreground's movement. No scale changing for the foreground is considered in this paper.

In section two we explain our video inpainting method. Then in section three we apply the proposed method to a video and compare our results with another video inpainting result. In section four we express conclusion and some conditions that can be considered in later studies.

II. The Proposed Video Inpainting Method

We have developed a new video inpainting approach for filling the hole created by an object. This approach applies contours to better fill in the missing data. Some weaknesses of the video inpainting methods which are usually applied todays are ghost shadows and running time. Fig. 1 shows a schematic overview of this algorithm.

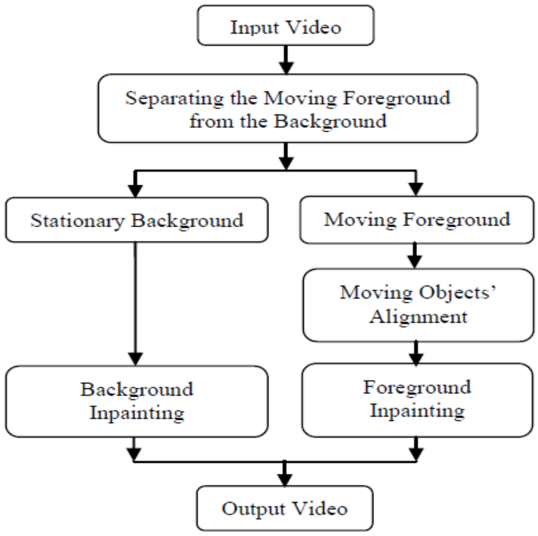


Figure 1. Schematic overview of the proposed algorithm

First, by using a mechanism based on thresholding, the moving foreground and the stationary background are separated from each other. Then each of these segments is inpainted individually. Finally, the inpainted background and the foreground frames are composed; so, the video is reconstructed.

A. Separating the Moving Foreground from the Background

In order to separate the moving foreground from the static background in each frame, a threshold value is used. This value is applied on the absolute difference between the current frame and the modeled background. Equation (1) is for this purpose.

$$F_t = \begin{cases} 1 & if\ |Img_t - B_{t-1}| > T \\ 0 & otherwise \end{cases} \quad (1)$$

where Img_t is the input frame at time t, $B_{t\text{-}1}$ is the background model, and F_t is the foreground at frame t. Gaussian Mixture Model (GMM), originally introduced in [14] and improved in [15] is used to model the background. In this approach, each pixel is modelled as a mixture of L Gaussian distributions. The parameters of the model are constantly updated. For each pixel, this approach automatically selects the number of components of GMM. The probability of observing the pixel at time t with this assumption that the observation of a pixel at previous time is given, is

$$P(X_t|B) \cong \sum_{l=1}^{L} w_{l,t} \mathcal{N}(X_t, \mu_{l,t}, \Sigma_{l,t}) \quad (2)$$

where L is the number of parameters and $w_{l,t}$, $\mu_{l,t}$ and $\sum_{l,t}$ are the computation of weight, mean, and covariance matrix of the i^{th} Gaussian in the mixture at time t, respectively. Modelling of each pixel is done, so the background can be estimated by the K largest components of the mixture model where K is given from equation (3).

$$K = \underset{k}{argmin}\left(\sum_{i=1}^{k} w_i > (1-c)\right) \quad (3)$$

where c is the maximum part of data that can be introduced as foreground. As explained before, difference of each frame from the modeled background recognizes the moving foreground. Background subtraction is done for pixels. After this, a median filter is used for each acquired foreground frame. This filter removes the outlier pixels.

Background subtraction is challenging for computer vision tasks. The focus of our work is to introduce a better algorithm for inpainting. Since we assumed the background is static in our video sequence, the foreground/background segmentation approach which was mentioned above is suitable for our work. More advanced methods for foreground/background segmentation can be used if necessary. If the background is dynamic, there are other methods for background subtraction such as the one presented in [16].

B. Moving Objects' Alignment

The moving object and the background are segmented. Now, a rectangular window is used to define the boundary of each object. This window is placed around the object. Since the moving object in our case has different states for motion, different window sizes are acquired. By choosing maximum size of the height and width for windows, a reference window is defined. The object is placed in the center of the reference window. By alignment of reference windows that cover the moving object, a mosaic image is acquired. There may be some possible overlap between objects, so to remove this, [13] changed the center of each object that is defined as:

$$c'_i = c'_{i-1} + w + \Delta c_i, \quad \Delta c_i = c_i - c_{i-1} \quad (4)$$

where c'_i is the center of object which is at frame i in the mosaic image, w is the width of reference window, and Δc_i is the distance between the centers of the objects at two frames that are consecutive.

Determination of the first and the last corrupted frames is done manually. The object center in corrupted frames is missed, so the velocity of the object which is shown by Δc is used to estimate the center of the object in these frames. In this approach there is no difference between partial occluded objects and completely occluded objects. So, only those frames which have a complete object are kept. Other frames that are either partially occluded or completely occluded are assumed to have no objects in them. In Fig. 2 the mosaic image of some of correct frames is shown.

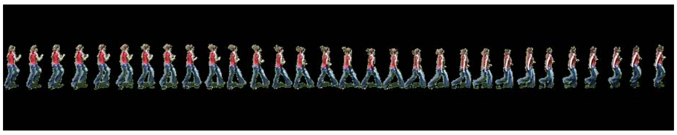


Figure 2. Mosaic image of some of correct frames

C. Background Inpainting

Background inpainting is done using exemplar-based inpainting which is proposed in [17].

The target region (the region which its data is missed) is filled by patches selected from the source region (the region which its data is accurate). The priority of the pixels in the boundary of the hole (the target region) is computed individually. The region centered at the pixel which has the biggest value for the priority is the first area to be completed. The priority and the order of patch selection are determined in accordance to structure of the image, entirely. The source region, target region and original image are denoted as Φ, Ω and I. To fill a distinct area named hole, the priority of each pixel that is placed on the boundary of the target region, $\delta\Omega$ is computed. To calculate the priority of a pixel named p, Equation (5) is used.

$$P(p)=C(p)D(p) \quad (5)$$

where $C(p)$ and $D(p)$ stand for confidence and data terms, respectively. These two terms are calculated as follows:

$$C(p) = \frac{\sum_{q \in \Psi_p \cap (I-\Omega)} C(q)}{|\Psi_p|} \quad (6)$$

$$D(p) = \frac{|\nabla I_p^{\perp} \cdot n_p|}{\gamma} \quad (7)$$

where Ψp is a patch with mid-point on location p, $|\Psi p|$ is the area of this patch, γ is a normalization factor (e.g. for greyscale images, γ =255). n_p is orthogonal unit vector to the boundary of the hole (target region), and $\perp$ expresses the orthogonal operator.

After sorting the priorities, the patch $\Psi p'$ which has the maximum priority, is found. Then the patch $\Psi q'$ which is the best matching one in the source region, is selected. It means the patch $\Psi q'$ minimizes the Sum Square Differences (SSD). After selecting $\Psi q'$, the patch is copied to $\Psi p'$. After all, confidence values for all pixels of the patch $\Psi p'$ which intersect with the target region are updated. This algorithm is applied iteratively till the hole becomes filled.

The assumption in our work is a stationary background. Because of this, the missing region is the same for all frames. So, there is no frame history for the hole and the information of the current frame is what is used in inpainting the hole. The exemplar-based method which maintains the structure and texture information, is an appropriate method to fill the hole. If the background is moving, not stationary, there is a frame history that can be used to complete the hole which is in the current frame. Inpainting moving backgrounds compared to static backgrounds show more discontinuation. So, filling the moving background is more challenging compared to filling the static background.

D. Foreground Inpainting

The main step in this study is the completion of missing frames in a mosaic image. A contour-based foreground comparison and a patch-based image inpainting algorithm are used for completion of the acquired mosaic image.

To have a simple explanation, we considered to have one moving object. Object motion is assumed to be periodic without any changes for scale. Also, the occluding object is not that big to destroy all motion periods (i.e., at least a complete motion period is kept). The patch here is considered to be large enough to enclose an object entirely. This assumption (large size for patches) is because of preserving structure and temporal continuity of the moving object.

In our study, the height and the width of the patch are considered to be the same as the height and the width of the reference window. As mentioned before, no partially occluded object is kept for the mosaic image; so, there only exists some complete objects and some completely occluded objects. Because of this, there is no difference to start from the left side of the mosaic image or the right side of it. If the object sequence is defined as $o_1 o_2 \ldots o_h o_{h+1} \ldots o_{h+n+1} \ldots o_N$, and the period of the motion is per, then $o_i \approx o_{i+per}$ (i indicates the frame number and o_{h+1} to o_{h+n} are objects in the damaged region). To select the most suitable patch, absolute difference between the contours of all complete patches and the object right one before the hole is calculated (here, completion is done from left side of the hole). This absolute difference is defined as $d(con_t, con_h)$, where con stands for contour. Moreover, to preserve continuity, absolute difference of contours one right before the hole and one right after the hole are participated in error calculation. It means first we calculate the difference between the contour one before the hole (con_h) with any contours in contour database. Once the best one is found (con_b), we calculate the difference between $con_{b\text{-}1}$ with $con_{h\text{-}1}$ and con_{b+n} with con_{h+n+1}. This iterates for five times (for five best candidates of con_bs) and the error is summed over three error calculation per iteration. The contour with the minimum overall error is the best one to complete first corrupted frame. Fig. 3 shows a part of mosaic image consisting of contours of some not occluded objects.

Since the motion is periodic, the remaining damaged frames can be filled by the frames subsequently located after the selected patch on the mosaic image. Besides of keeping periodicity of the motion, the time spent to find the best suitable patch is reduced in this study compared to other works. This is firstly because of contour-based comparing method which reduces time spending to find the patch with minimum error.

Moreover, the assumption of periodic motion helps us to copy other patches without spending time to calculate absolute difference. In the case of non-periodic motions for objects, this approach will not be useful. To fill the missing region in the presence of non-periodic movements, sum squared error should be calculated for each of the holes, individually.

Finally, to construct the output video, objects of different frames are separated from the mosaic image.

Figure 3. Mosaic image of some not occluded objects consisting of contours

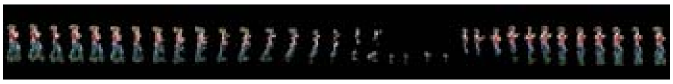
By locating the objects on their places, the frames of the inpainted video sequence are obtained.

III. Experimental Results

The video used in our task is the "jumping girl" that was captured by [3]. In this video, a girl moves from left to right. There is a person who acts as an occluding object and the girl passes behind this person. Fig. 4 shows the results of our algorithm on this video.

Fig. 5 shows the difference between the results of the proposed algorithm and those of the space-time completion of video ([3]). Over-smoothing can be seen on results of [3].

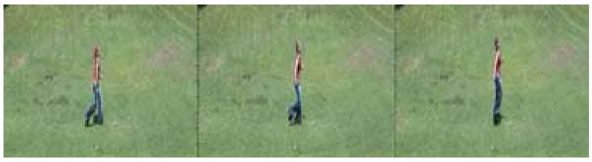
By comparing these two groups of images, it can be find out that our algorithm represents better results. Since the proposed algorithm distinguishes between the moving foreground and the static background, there is not any more over-smoothing on the background. Video can show existence or not-existence of over-smoothing better than images; so, to better comparison you can see our video at *https://rapidshare.com/files/1719767656/jumping_girl_inpainted.rar*. Time spent to find the best patch is reduced because of using the contour-based approach. But there is a jump when the moving object enters or exits the hole.



(a)

(b)

Figure 4. Results of the proposed algorithm (a) Original mosaic image in presence of an occluding object; (b) Mosaic image of inpainted foreground



(a)

(b)

Figure 5. Comparison of the results of our algorithm with the one proposed in [3] (a) Results of algorithm proposed in [3]; (b) Results of our algorithm

Note that here the object's motion is assumed to be periodic that helps filling the large holes by the use of consecutive frames.

IV. CONCLUSION AND FUTURE WORK

This paper presented a new method for inpainting videos. First of all, the moving object and the background are separated from each other. Then a mosaic image is constructed. To fill in the damaged frames, a patch-based image inpainting method and a contour-based comparison method are used. The patches are defined large enough to enclose the whole object. After this step, the objects are settled on their locations on the foreground. Finally, these placements are superimposed with the inpainted background.

The results of this proposed algorithm is acceptable except for videos which have non-stationary backgrounds. The main challenge in this approach is to reduce the running time while the quality of video inpainting is kept.

Still some limitations remain that can be solved. In the case of non-periodic motion, the proposed method will no longer produce pleasant results. Better methods for detecting contours will end to better completion. Considering partially occluded objects may solve the jump problem which occurs on the entrance or exodus the hole. More challenging approaches include considering dynamic background, camera movement or scale changing of the foreground object.

REFERENCES

[1] N. C. Tang, T. K. Shih, "Digital Inpainting and Video Falsifying", *In Proc. IEEE Int. Symposium on Information Technology(ITSim)*, pp. 1-8, 2008.

[2] R. Rodrigo, Z. Chen, J. Samambandu, "Energy Based Video Synthesis", *In Proc. 5th WSEAS Int. Conf. Telecommunications and Informatics*, pp. 198-202, 2006.

[3] Y. Wexler, E. Shechtman, M. Irani, "Space-Time completion of video", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 463-476, 2007.

[4] M. V. Venkatesh, S. S. Cheung, J. Zhao, "Efficient Object Based Video Inpainting", *Journal of Pattern Recognition Letters*, vol. 30, no. 2, pp. 168-179, 2009.

[5] M. Bertalmio, A. L. Bertozzi, G. Sapiro, "Navier-Strokes, Fluid Dynamics, and Image and Video Inpainting", *In Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 355-362, 2001.

[6] K. A. Patwardhan, G. Sapiro, M. Bertalmio, "Video Inpainting under Constrained Camera Motion", *In Proc. IEEE Trans. Image Processing*, vol. 16, no. 2, pp. 545-553, 2007.

[7] T. K. Shih, N. C. Tang, J. N. Hwang, "Ghost Shadow Removal in Multi-Layered Video Inpainting", *IEEE Int. Conf. Multimedia and Expo.*, pp. 1471-1474, 2007.

[8] T. Ding, M. Snaier, O. I. Camps, "A Rank Minimization Approach to Video Inpainting", *In Proc. 11th IEEE Int. Conf. Computer Vision*, pp. 1-8, 2007.

[9] R. C. Chang, N. C. Tang, C. C. Chao, "Application of Inpainting Technology to Video Restoration", *In Proc. 1st IEEE Int. Conf. Ubi-Media Computating*, pp. 359-364, 2008.

[10] T. K. Shih, N. C. Tang, J. N. Hwang, "Exemplar-Based Video Inpainting Without Ghost Shadow Artifacts by Maintaining Temporal Continuity", *In Proc. IEEE Trans. Circuits and Systems for Video*, vol. 13, no. 3, pp. 347-360, 2009.

[11] N. C. Tang, H. Y. M. Liao, C. W. Su, F. Huang, T. K. Shih, "Video Inpainting on Digitized Old Films", *In Proc. 13th Int. Conf. Knowledge-Based and Intelligent Information and Engineering Systems*, vol. 5712, pp.421-430, 2009.

[12] C. W. Lin, N. C. Cheng, "Video Background Inpainting Using Dynamic Texture Synthesis", *In Proc. IEEE Int. Symposium on Circuits and Systems*, pp. 1559-1562, 2010.

[13] A. Koochari, M. Soryani, "Exemplar-Based Video Inpainting with Large Patches", *Journal of Zhejiang University*, vol. 11, no. 4, pp. 270-277, 2010.

[14] C. R. Wren, A. Azarbayejani, T. Darrell, A. P. Pentland, "Pfinder: Real-Time Tracking of the Human Body", *IEEE Trans. Pattern Analysis*, vol. 19, no. 7, pp. 780-785, 1997.

[15] Z. Zivkovik, "Improved Adaptive Gaussian Mixture Model for Background Subtraction", *Proc. 17th Int. Conf. on Pattern Recognition*, vol. 2, pp. 28-31, 2004.

[16] Y. Sheikh, M. Shah, "Bayesian Object Detection in Dynamic Scenes", Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol.1, pp. 74-79, 2005.

[17] A. Criminisi, P. Perez, K. Toyama, "Region Filling and Object Removal by Exemplar-Based Inpainting", *IEEE Trans. Image Processing*, vol. 13, no. 9, pp. 1200-1212, 2004.