Video Completion via Motion Guided Spatial-Temporal Global Optimization

Ming Liu[1], Shifeng Chen[2], Jianzhuang Liu[1,2], and Xiaoou Tang[1,2]
[1]Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong
[2]Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China
{mliu6, jzliu, xtang}@ie.cuhk.edu.hk sf.chen@sub.siat.ac.cn

ABSTRACT

In this paper, a novel global optimization based approach is proposed for video completion whose target is to restore the spatial-temporal missing regions of a video in a visually plausible way. Our algorithm consists of two stages: motion field completion and color completion via global optimization. First, local motions within the missing parts are completed patch-by-patch greedily using pre-computed available motions in the video. Then the missing regions are filled by sampling patches from available parts of the video. We formulate the video completion as a global energy minimization problem by Markov random fields (MRFs). Based on the completed motion field of the video, a well-defined energy function involving both spatial and temporal coherence relationship is constructed. A coarse-to-fine Belief Propagation (BP) is proposed to solve the optimization problem. Experimental results have demonstrated the good performance of our algorithm.

Categories and Subject Descriptors

I.4.9 [**IMAGE PROCESSING AND COMPUTER VISION**]: Applications

General Terms

Algorithms, Experimentation

1. INTRODUCTION

Image and video completion, also known as image and video inpainting, are of great importance in many multimedia applications such as photo and movie post-production. Their goal is to automatically reconstruct missing regions in an image/video in a non-detectable form. A number of methods have been proposed to deal with the problem of image completion [2], [3], [6], [8], [13]. A partial differential equation (PDE) based algorithm is presented in [2]. Inspired by the texture synthesis technique in [4], an exemplar-based technique [3] is proposed to repair the missing regions via patch copying. The global optimization based algorithms [6], [8], [13] can overcome the shortcoming of the greedy patch copying scheme in [3].

Compared with image completion, video completion is more challenging in two aspects. First, it is more important to enforce temporal coherency than spatial coherency in the completion process since human visual system is more sensitive to motion distortion. Simply treating video as a set of independent images and then applying an image completion method to them are not advisable. Second, video completion contains much more data and thus needs more efficient algorithms.

One of the first efforts for video completion is made in [1], which is a PDE-based approach and handles the video frame by frame. It works well in small structured holes, but fails to complete large holes in a video sequence and does not utilize the temporal information from the video. Many segmentation based or layer extraction based algorithms are developed recently [7], [15], [10], [9]. In addition to being restricted to periodic motion, these methods have at least one of the following limitations: large computational complexity, interaction requirement, and inaccurate layer extraction.

Extending the exemplar-based approach to video completion, the algorithm in [14] treats video completion as a global optimization problem. However, the algorithm also relies on the assumption of periodic motion and is computationally inefficient due to the pixel-by-pixel filling process and exhaustive search for candidates.

A newly published algorithm in [12] restores local motion in the holes of the video by sampling spatial-temporal motion patches, instead of directly using the color copy-and-paste scheme. With the completed motion volume, color is propagated into the holes to complete the video. As discussed in [12], the algorithm is more sensitive to noise than directly using color sampling and does not work well for the completion of videos with large motions. Moreover, the results of this algorithm have blurring effects due to the weighted average scheme in color propagation.

In this paper, we propose a motion guided spatial-temporal global video completion algorithm to combine motion field completion and global exemplar-based color completion. First, the motion in the data missing regions is completed patch-by-patch using the motion pre-computed in the available regions. Then based on the completed motion, exemplar-based color completion is formulated as a discrete global optimization problem with a well defined objective function, which enforces spatial and temporal consistency under the Markov random field (MRF) model. A coarse-to-fine belief propagation (BP) is proposed to deal with the intolerable computational cost caused by the large number of label candidates in the optimization.

Our algorithm preserves the temporal consistency information based on the completed motion field, and globally optimizes the color completion process. Moreover, our algorithm is not restricted to videos containing periodic motion only and can handle a wide variety of videos.

MM'09, October 19–24, 2009, Beijing, China.

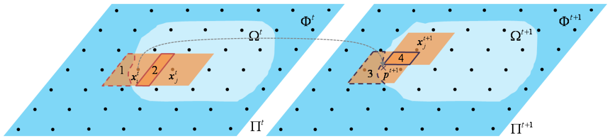

Figure 1: Illustration of the spatial and temporal terms. The dots indicate the sampled pixels which correspond to the vertices in the graph. Regions 1, 2, 3, and 4 are overlapping parts for the calculation of $E_1(x_i^t)$, $E_2(x_i^t, x_j^t)$, $E_3(x_i^t)$, and $E_4(x_i^t, x_j^{t+1})$, respectively. The patch centered at p^{t+1} (the cross) is copied from x_i^t.

2. MOTION GUIDED SPATIAL-TEMPORAL GLOBAL OPTIMIZATION

We formulate the video completion problem as a labeling problem modeled by discrete Markov Random Fields (MRFs). Let $f = \{f^t\}_{t=1}^T$ be the input video of T frames with the region $\Pi = \{\Pi^t\}_{t=1}^T$, where Π^t is the region of f^t. Suppose that $\Phi = \{\Phi^t\}_{t=1}^T$ is the source region and $\Omega = \{\Omega^t\}_{t=1}^T$ is the target region (data missing region). Then we have $\Phi + \Omega = \{\Phi^t + \Omega^t\}_{t=1}^T = \{\Pi^t\}_{t=1}^T = \Pi$.

Firstly, we sparsely sample each frame with a horizontal spacing hs and vertical spacing vs. Then we can obtain sampled pixels $P = \{\{p_i^t\}_{i=1}^{N^t}\}_{t=1}^T$ in the target region, where N^t is the number of sampled pixels in the target region of the tth frame. The process of video completion is to fill the target region by pasting some $w \times h$ patches taken from the source region to the locations centered at the positions in P.

We construct an undirected weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the node set $\mathcal{V} = \{\{v_i^t\}_{i=1}^{N^t}\}_{t=1}^T$ contains all the pixels in P, and $\mathcal{E}$ is the set of edges connecting each node to nodes in its neighborhood system. A 4-neighborhood system is used to enforce the *spatial consistency* constraint in the same frame, while some nodes in sequential frames, called *temporal neighbors*, are included in our neighborhood system to enforce the *temporal consistency* constraint. The detail of temporal neighbors is described in Section 2.2.2.

Let $\mathcal{L} = \{l_k\}_{k=1}^K$ be the set of label candidates containing all the $w \times h$ patches taken from the source region. Our labeling problem is to find the best label configuration $X = \{\{x_i^t\}_{i=1}^{N^t}\}_{t=1}^T$ such that an energy function is minimized, where $x_i^t \in \mathcal{L}$ and $x_i^t = l_k$ represents that the label (patch) for node v_i^t is l_k. In our approach, the best label configuration is estimated by minimizing the following energy function:

$$E(X) = E_s(X) + \alpha E_t(X), \tag{1}$$

where $E_s(X)$, called *spatial term*, enforces the spatial consistency constraint, $E_t(X)$, called *temporal term*, enforces the temporal consistency constraint, and α is a positive constant to balance these two terms. Fig. 1 illustrates the spatial and temporal terms.

2.1 The Spatial Term

The spatial term implies that the overlapping parts of patches should have consistent texture and structure information in the patch pasting process. Based on the MRF model, it is defined as:

$$E_s(X) = \sum_{v_i^t} E_1(x_i^t) + \sum_{(v_i^t, v_j^t) \in \mathcal{E}_s} E_2(x_i^t, x_j^t), \tag{2}$$

where $\mathcal{E}_s$ is the spatial 4-neighborhood system, $E_1(x_i^t)$ is the cost for label x_i^t, and $E_2(x_i^t, x_j^t)$ is the consistency cost for label pair (x_i^t, x_j^t).

Similar to [3], the *confidence map* is also used in our algorithm to represent the importance of nodes in the filling process. In the map, the pixels in the target region closer to the source region in each frame have larger confidence values. With the confidence map, the cost for label x_i^t is defined as:

$$E_1(x_i^t) = C_i^t \cdot d(x_i^t, \Phi^t), \tag{3}$$

where C_i^t is the confidence value for node v_i^t and $d(x_i^t, \Phi^t)$ constrains the synthesized patch x_i^t to match well with the source region which overlaps with the node v_i^t. $d(x_i^t, \Phi^t)$ is calculated as the sum of the squared differences (SSD) of the pixel colors in the overlapping part between x_i^t and Φ^t (e.g., region 1 surrounded by the red dashed curve in Fig. 1). When x_i^t and Φ^t do not overlap, $E_1(x_i^t) = 0$.

Since structure (e.g., lines, curves) continuity is important for human perception and texture reflects the details of an image, we incorporate both structure and texture in the completion process. The consistency cost $E_2(x_i^t, x_j^t)$ in (2) is thus defined as

$$E_2(x_i^t, x_j^t) = \left[\frac{C_i^t + C_j^t}{2}\right]\left[\lambda_1 E_2'(x_i^t, x_j^t) + \lambda_2 E_2''(x_i^t, x_j^t)\right], \tag{4}$$

where C_i^t and C_j^t are the confidence values of nodes v_i^t and v_j^t, respectively, $E_2'(x_i^t, x_j^t)$ is used to enforce consistency for texture propagation, $E_2''(x_i^t, x_j^t)$ is for structure propagation, and λ_1 and λ_2 are two factors to balance E_1, E_2', and E_2''.

In our algorithm, $E_2'(x_i^t, x_j^t)$ is computed by

$$E_2'(x_i^t, x_j^t) = d(x_i^t, x_j^t), \tag{5}$$

where $d(x_i^t, x_j^t)$ is the SSD in the overlapping part between the patches centered at nodes v_i^t and v_j^t (e.g., region 2 surrounded by the red solid curve in Fig. 1). $E_2''(x_i^t, x_j^t)$ is computed by

$$E_2''(x_i^t, x_j^t) = d_{gh}^2(x_i^t, x_j^t) + d_{gv}^2(x_i^t, x_j^t), \tag{6}$$

where $d_{gh}(x_i^t, x_j^t)$ and $d_{gv}(x_i^t, x_j^t)$ are the gradient differences between x_i^t and x_j^t in the image horizontal and vertical directions, respectively. The gradient of a patch is denoted as the maximum gradient of the pixels in the patch, which describes the structure of the patch. The constraint of gradient consistency propagates the structure information.

2.2 The Temporal Term

The temporal term constrains that two corresponding patches in two sequential frames should have consistent colors. In our algorithm, the correspondence is found via local motion estimation. The hierarchical Lucas-Kanade algorithm [11] is used for motion estimation.

If dense motion is estimated, the correspondences for all patches in a video without missing pixels can be constructed. In our problem, however, there are many data missing regions in the input video, and thus optical flows cannot estimate the motions for the pixels in these regions. To obtain a completed motion map, we first calculate the motions for all the pixels in the source region. Then copy-and-paste scheme is used to carry out the motion completion. The details are described as follows.

2.2.1 *Motion Completion*

In our approach, a copy-and-paste process, i.e., copying the best motion patch from the source region and pasting it to the target region, fills in the motion in the target region patch by patch.

Before defining the criteria for choosing the best source patch for a target region, the motion difference measurement is introduced.

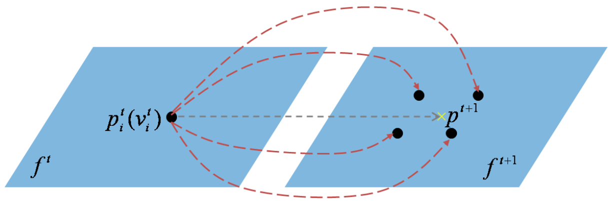

Figure 2: Illustration of the temporal neighborhood system. p_i^t is a sampled pixel in frame t with its corresponding graph vertex v_i^t. The cross in frame $t+1$ is the corresponding position of p_i^t based on the motion estimated. Then vertices corresponding to the four nearest sampled pixels in frame $t+1$ are the temporal neighbors of v_i^t (connected with v_i^t by red dashed lines).

Suppose that the motion vector of pixel q in frame t is $(u_q^t, v_q^t)^T$. If we regard the 2D motion as a 3D vector in the spatio-temporal domain by padding the element t, the 3D vector is defined as $\mathbf{m}_q^t = (u_q^t, v_q^t, t)^T$. The difference between two motion vectors $\mathbf{m}$ and $\mathbf{m}'$ is defined as the angular difference [12]:

$$d_m(\mathbf{m}, \mathbf{m}') = 1 - \frac{\mathbf{m} \cdot \mathbf{m}'}{|\mathbf{m}||\mathbf{m}'|} = 1 - \cos\theta, \tag{7}$$

where θ is the angle between the two motion vectors $\mathbf{m}$ and $\mathbf{m}'$.

For a source motion patch A_s and a target motion patch A_t (A_s and A_t are 3D in the spatio-temporal domain), the difference measurement between them is defined as:

$$d_{mp}(A_s, A_t) = \frac{1}{|Q_s|} \sum_{q_t \in Q_s} d_m(\mathbf{m}_{q_s}^{t_s}, \mathbf{m}_{q_t}^{t_t}), \tag{8}$$

where Q_s is the set of points in A_t belonging to the source region, $|Q_s|$ is the number of pixels in Q_s, q_s and q_t are two corresponding pixels in A_s and A_t respectively, and t_s and t_t are the frames in which A_s and A_t are respectively. Then for A_t the best source patch $\widehat{A}_s$ is chosen by minimizing (8):

$$\widehat{A}_s = \operatorname*{argmin}_{A_s} d_{mp}(A_s, A_t). \tag{9}$$

2.2.2 A Temporal Energy Function

Before defining the temporal term, the temporal neighborhood is introduced first. For a sampled pixel p_i^t whose corresponding graph vertex is v_i^t, if its motion is known, then we can find its corresponding point p^{t+1} in the next frame. We call the set of the four vertices in frame $t+1$ corresponding to the four sampled vertices nearest to p^{t+1} the *temporal neighborhood* of v_i^t (see Fig. 2). If v_j^{t+1} is a temporal neighbor of v_i^t, then we denote them as $(v_i^t, v_j^{t+1}) \in \mathcal{E}_t$.

The definition of the temporal term is similar to the spatial term, which is expressed as the sum of two parts:

$$E_t(X) = \sum_{v_i^t} E_3(x_i^t) + \sum_{(v_i^t, v_j^{t+1}) \in \mathcal{E}_t} E_4(x_i^t, x_j^{t+1}), \tag{10}$$

where $\mathcal{E}_t$ is the temporal neighborhood system, $E_3(x_i^t)$ represents the temporal inconsistency between x_i^t and its corresponding source region in frame $t+1$, and $E_4(x_i^t, x_j^{t+1})$ represents the temporal inconsistency between x_i^t and x_j^{t+1}.

The definitions of $E_3(x_i^t)$ and $E_4(x_i^t, x_j^{t+1})$ are similar to those of $E_1(x_i^t)$ and $E_2(x_i^t, x_j^t)$, respectively, but compared with $E_1(x_i^t)$ and $E_2(x_i^t, x_j^t)$, there is no confidence and structure information in $E_3(x_i^t)$ and $E_4(x_i^t, x_j^{t+1})$. They are defined as:

$$E_3(x_i^t) = d(x_i^t, \Phi^{t+1}), \quad E_4(x_i^t, x_j^{t+1}) = d(x_i^t, x_j^{t+1}). \tag{11}$$

As in the spatial term, here d is the SSD value in the overlapping region of the two parts. Suppose that the corresponding pixel of p_i^t in frame $t+1$ is p^{t+1}. To calculate $d(x_i^t, \Phi^{t+1})$ and $d(x_i^t, x_j^{t+1})$, the first step is to put the center of the patch x_i^t at p^{t+1}. Then $d(x_i^t, \Phi^{t+1})$ is the SSD value in the overlapping region between the patch and Φ^{t+1} (e.g., region 3 surrounded by the purple dashed curve in Fig. 1), and $d(x_i^t, x_j^{t+1})$ is the SSD value in the overlapping region between the patch and x_j^{t+1} (e.g., region 4 surrounded by the purple solid curve in Fig. 1).

2.3 Optimization by BP

The problem of minimizing (1) is NP-hard. BP can find a local optimum for such an MRF energy function. The max-product and sum-product are two typical BP algorithms [5]. In this paper, the max-product algorithm is used since it is less sensitive to numerical inaccuracy.

The max-product BP works iteratively by passing messages along the graph. For a graph with N nodes and K label candidates, the running time for T iterations is $O(TNK^2)$. In our video completion approach, the main problem with such a standard BP algorithm is that the number of label candidates K is too large to be used in practice. In this paper, we use a coarse-to-fine scheme to greatly reduce the computational time. The main idea of this scheme is to perform BP R times with K_r label candidates each time, $r = 1, ..., R$, instead of running BP once with K candidates, where K_r is much smaller than K.

For simplicity, we take a two-layer pyramid as an example to explain the scheme. Let K_1 and K_2 be the numbers of candidates in the first and the second BP executions respectively. We first use the k-means algorithm to classify all the patches in $\mathcal{L}$ into K_1 clusters, denoted as $S_1, S_2, ..., S_{K_1}$, i.e., $\mathcal{L} = \{S_1, S_2, ..., S_{K_1}\}$. The first running of BP takes the K_1 cluster centers as the label candidates $\mathcal{L}^1 = \{c_1, c_2, ..., c_{K_1}\}$ to find the best label configuration $X_1 = \{x_1^1, x_2^1, ..., x_N^1\}$ that minimizes the objective energy function, where $x_i^1 \in \mathcal{L}^1$, $1 \leq i \leq N$. Then we perform BP again. Suppose that after the first BP, the best label for node v_i is $x_i^1 = c_{k_1}$. In the second round BP, the new label candidates for node v_i are all the elements[1] belonging to the cluster with center c_{k_1}. Using such different label candidate sets for different nodes, the second BP runs to find the best label configuration.

Obviously, such a coarse-to-fine BP scheme leads to a result different from that obtained with the BP running once. However, our experiments show that this scheme can achieve satisfactory results. The most important benefit of this scheme is that it can make our algorithm practical. This scheme can also be used to speed up some other MRF based applications in computer vision and graphics.

3. EXPERIMENTS

In our experiments, we validate our algorithm on various videos representing different interesting and challenging cases to demonstrate its effectiveness. Due to the space limitation, we only show a few selected results from four representative videos, 120-frame "performance" (180 × 240) [12], 88-frame "beach" (80 × 170) [14], 40-frame "running" (240 × 320) [9], and 19-frame "car" (240 × 320) [9]. For all our experiments, the parameters in our algorithm are chosen as $\lambda_1 = 1$, $\lambda_2 = 1.5$, and $\alpha = 5$. The number of levels R in the multi-level BP is chosen as 2 or 3, depending on the size of a video.

Fig. 3 shows the results for the video "performance". The first row gives 4 original frames. We want to remove the walking spec-

[1] To limit the maximum label candidate number, if the number is larger than K_2, K_2 candidates are randomly selected.

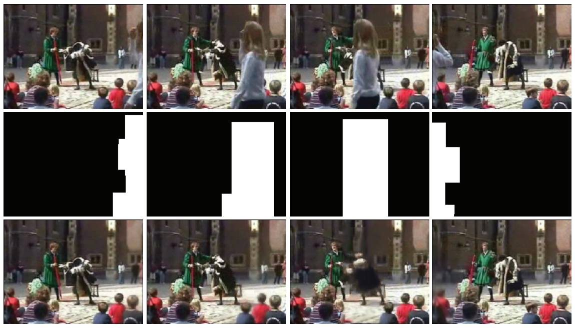

Figure 3: Some results on the "performance" video. The three rows show the original frames, the manually removed regions, and the video completion results by our algorithm, respectively.

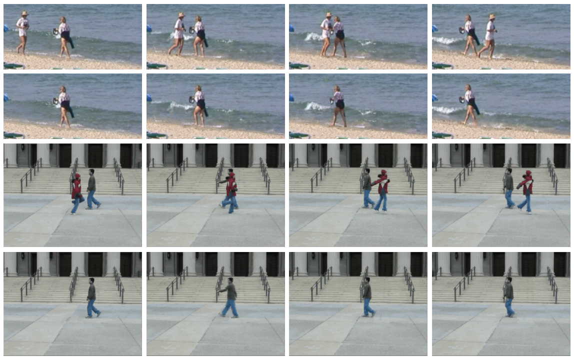

Figure 4: Some results on the "beach" and "running" videos.

tator. The second row shows the manually removed regions roughly covering the spectator. The last row displays the completion results by our algorithm. Fig. 4 and Fig. 5 show the other results, without the manually removed regions given due to the space limitation.

As shown in Fig. 3, the spectator takes a large space in each frame, and non-periodic motion happens in this video. The approach in [9], therefore, cannot handle this video completion well due to its periodic motion constraint and the large data missing. Another recent algorithm [12] leads to serious blurring results for this video, as stated in [12], because of its simple weighted average scheme in color propagation. However, our algorithm generates promising results on this challenging case. In the "running" video, the camera taking the video is also moving. Our algorithm can fill in the holes well. Another challenging case in video completion is to complete the regions where the sizes of the objects change. Fig. 5 is such an example where the car moves closer to the camera. Our algorithm is still successful to complete the removed sign post.

From the experimental results, we can see that our algorithm can handle a variety of video completion tasks with different situations, such as dynamic foreground and background, camera motion, object scale changing, and large data missing. Besides, there is no periodic motion restriction imposed on our algorithm.

4. CONCLUSION

In this paper, a novel video completion algorithm has been proposed by combining motion completion and global exemplar-based color completion. For a video with holes, the motion field in the holes is filled locally first. Based on the completed motion field, color is restored in a global exemplar-based scheme by minimizing an MRF energy function. The proposed objective function enforces both spatial and temporal consistency constraints in the color completion process. BP is used to solve the minimization problem. To avoid the computational impracticability caused by the large number of label candidates in BP optimization processing, we utilize a coarse-to-fine optimization scheme whose essential idea is to carry out BP multiple times with sharply reduced number of label candidates, instead of running BP once with a large number of label candidates. The experimental results on a variety of videos have demonstrated the good performance of our approach.

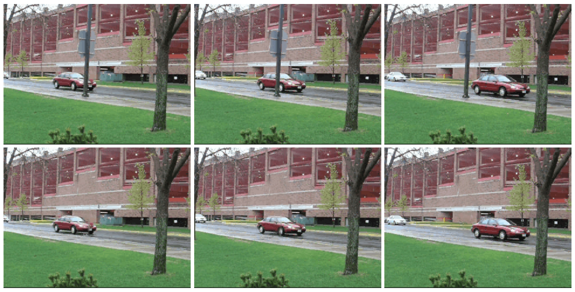

Figure 5: Some results on the "car" video.

5. ACKNOWLEDGMENTS

This work was supported by grants from Microsoft Research Asia, the Research Grants Council of the Hong Kong SAR, China (Project No. CUHK 414306, 415408), and Shenzhen Bureau of Science Technology & Information, China.

6. REFERENCES

[1] M. Bertalmıo, A. Bertozzi, and G. Sapiro. Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting. *CVPR*, pages 355–362, 2001.

[2] M. Bertalmıo, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. *SIGGRAPH*, pages 417–424, 2000.

[3] A. Criminisi, P. Perez, and K. Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. on Image Processing*, pages 1200–1212, 2004.

[4] A. Efros and T. Leung. Texture synthesis by non-parametric sampling. *ICCV*, pages 1033–1038, 1999.

[5] P. Felzenszwalb and D. Huttenlocher. Efficient Belief Propagation for Early Vision. *IJCV*, pages 41–54, 2006.

[6] T. Huang, S. Chen, J. Liu, and X. Tang. Image inpainting by global structure and texture propagation. *ACM MM*, 2007.

[7] J. Jia, T. Wu, Y. Tai, and C. Tang. Video repairing: Inference of foreground and background under severe occlusion. *CVPR*, pages 364–371, 2004.

[8] N. Komodakis and G. Tziritas. Image completion using global optimization. *CVPR*, pages 442–452, 2006.

[9] K. Patwardhan, G. Sapiro, and M. Bertalmio. Video Inpainting Under Constrained Camera Motion. *IEEE Trans. on Image Processing*, pages 545–553, 2007.

[10] Y. Shen, F. Lu, X. Cao, and H. Foroosh. Video Completion for Perspective Camera Under Constrained Motion. *ICPR*, pages 63–66, 2006.

[11] J. Shi and C. Tomasi. Good features to track. *CVPR*, 1994.

[12] T. Shiratori, Y. Matsushita, S. Kang, and X. Tang. Video Completion by Motion Field Transfer. *CVPR*, 2006.

[13] J. Sun, L. Yuan, J. Jia, and H. Shum. Image completion with structure propagation. *SIGGRAPH*, pages 861–868, 2005.

[14] Y. Wexler, E. Shechtman, and M. Irani. Space-time video completion. *CVPR*, 2004.

[15] Y. Zhang, J. Xiao, and M. Shah. Motion Layer Based Object Removal in Videos. *WACV/MOTIONS*, 2005.