A Fast Video Inpainting Technique

Mrinmoy Ghorai, Pulak Purkait, and Bhabatosh Chanda

Indian Statistical Institute
Kolkata, India
{mgre04,pulak.isi}@gmail.com,chanda@isical.ac.in

Abstract. In this paper we present a fast video inpainting technique to infer the unknown information in the target region by maximizing box-based self-similarity and coherence measure. The video inpainting is already proposed in the literature and some of them are able to produce good quality results. However, the bottleneck of those algorithms is they are painfully slow. Here we fill the texture in the target region that preserves the smooth motion of the object without inclusion of any artifacts in reasonable amount of time. Our experiments show that the proposed method is quite efficient to synthesize unknown information in a video and comparable to the existing state-of-the-art methods. Moreover, proposed method is based on box filling and optimization is done on multiple scale using EM algorithm, and is computationally faster than the existing ones.

Keywords: video inpainting, self-similarity, spatio-temporal coherency.

1 Introduction

Video inpainting or completion is a method to fill-in the unknown regions in a video or image sequence. This is an extension to image inpainting problem in some sense where fill-in occurred in an unknown region in the place of the object we wish to remove from an input image. But here in video inpainting, target region placed in all the frames of the video at same location or continuous moving location. We can think the target region as a “hole” or “tunnel” in the video. In video inpainting techniques, the hole is filled by the known matter in such a way that it possess a smooth transition of the object having a motion and passing through the hole. Here we develop similar kind of method people use for image inpainting.

Approaches to image inpainting may be divided roughly into three categories: (i) partial differential equation (PDE) based approach for structure propagation, (ii) exemplar-based approach for texture synthesis and (iii) coherency-based approach for global consistency. PDE-based approaches [1] fill the target region of an image by diffusing the known data from the source region towards the interior of the target region. The exemplar-based approach infer unknown information by copying most similar patch from the source region and filling in the unknown region [2,3,4]. In the process of copying patches to the target region, a number of authors suggested to incorporate some spatial coherence in the texture synthesis process [5,6]. The basic idea of this type of approach is to assign each pixel in the target region based on the correspondence of the neighbouring pixels. Bugeau *et al.* [7] proposed a combination of three previously

P. Maji et al. (Eds.): PReMI 2013, LNCS 8251, pp. 430–436, 2013.

mentioned methods, namely texture synthesis, diffusion (PDE) and coherence in a single framework to take the advantages of all the methods. In this paper, we focus on exemplar based video inpainting method that extend exemplar-based image inpainting methods.

Wexler *et al.* [8] are first to explore region completion for video. They solved the video completion problem as optimization of global objective function using coherence structure. They used an iterative method for filling-in each pixel of the target region in multiple scale. Similar kind of work for repairing damaged video has been reported in [9]. Though this method is able to produce good results in difficult cases, their method involve combining different techniques making the process of inpainting very slow and complicated. The algorithm in [10,11] estimates the motion information for each pixel in the video frame in order to determine whether a pixel belongs to a moving object or belongs to static background accordingly inpaint the moving objects based on the information at stationary background. Their assumption was that the target region is much smaller than moving object size, having stationary background (static camera) which may not be the case in real scenario. This method is comparatively fast but not up-to that level.

In this paper, we extend the method of self similarity and coherency proposed in various image inpainting techniques to video inpainting and try to develop an algorithm which is computationally efficient compare to the existing methods. Here we suggest to

- model the problem of video inpainting as a energy minimization task;
- incorporate the strength of self-similarity along with coherency in a comprehensive framework;
- Reduce the computational cost by incorporating 3D-box based processing and filling instead of one pixel at a time.

Our algorithm try to approximate to a global optimization problem that combines two fundamental concepts of self-similarity and coherency.

The rest of the paper is organized as follows. In the following Section 2 describes proposed 3D box-filling based coherent texture synthesis procedures in detail and more implementation issues are discussed in Section 5. At the end, we show some experimental results in Section 4 where some practical evidences are also presented and then conclude with mentioning some pros and cons in Section 5.

2 Proposed Video Inpainting

Here we want to infer the unknown/target regions in a given video from the known/source region without introducing any artifacts. The challenge of this type of problem is to complete target regions of the video sequence in such a way that it allows moving objects to transmit smoothly through the target region. If the target region is smaller compared to the moving object then difficulty is much less than when the target region is quite larger than the moving object. Traditional image inpainting technique in general fails to generate temporal information of the moving object.

Let V and $T \subset V$ be a video and a target region either fixed or continuously moving throughout the sequence. Since video is a space-time volume, we define spatio-temporal

3D box $S_p(x,y,t)$ of size $(n \times n \times n)$ centering at the location of a pixel (x,y) in the t^{th} frame. Let us denote the set of all 3D box (full or partial) in the target region T by Δ and set of all box in the whole sequence V by Γ. We extend the texture synthesis technique proposed by [7] for video by finding the *correspondence map* $\tau : \Gamma \rightarrow \Gamma \setminus \Delta$ that associates the boxes from V to the boxes of known portions $V \setminus T$ such that

$$\tau(S_p) = \begin{cases} S_p, & \text{if } S_p \in \Gamma \setminus \Delta \\ S_q \in \Gamma \setminus \Delta, & \text{if } S_p \in \Delta \end{cases} \tag{1}$$

where S_q is a 3D box from known region, most similar to the box S_p of target region. If we can able to find out an efficient correspondence map τ that can map each box of the target region to known region, then we can fill the boxes in the target region by the mapped boxes in the source region. Thus we eventually solve the inpainting problem. Therefore in the proposed method we seek for the optimal relevant corresponding map.

2.1 Coherency and Self-similarity

In iterative texture synthesis or image inpainting, self-similarity is used to refine the assignment of patches or, more accurately, pixels in each iteration. The idea is to use the output at the previous iteration as input for the current iteration. Efros *et al.* [12] proposed texture synthesis for images in the terms of correspondence map τ. Note that here we use τ as a correspondence map of boxes $S_p \in \Delta$. In the proposed method, we assume that our target portions T of the given video V has self-similarity and coherency with known parts of the video. In other words, we wish to complete the target portions T with some new data $\widehat{T}$ such that resulting video $\widehat{V}$ has as much self-similarity and global visual coherence as the source portions $V \setminus T$. Therefore, we seek a solution of the following maximization problem

$$\widehat{\tau} = \max_{\tau}[Coherence(\widehat{V}\ ,\ V \setminus T)] \tag{2}$$

where τ is the correspondence map. The $Coherence(\widehat{V},\ V \setminus T)$ is the measure of self-similarity and global visual coherence, defined as

$$Coherence(\widehat{V},\ V \setminus T) = \sum_{S_p \in \Delta} s\,(S_p, \tau(S_p)) \tag{3}$$

where $s\,(S_p, \tau(S_p))$ is the similarity measure between box around the voxel p and corresponding mapped box around voxel q (where $S_q = \tau(S_p)$). A formalism similar to eq. (3) was already used in [8] for summarizing visual data. In our case we have considered the similarity measure as:

$$s\,(S_p, \tau(S_p)) = \exp\left(\frac{-d_{SSD}\,(S_p,\ \tau(S_p))}{2\sigma^2}\right), \tag{4}$$

where $d_{SSD}\,(S_p,\ \tau(S_p))$ computes the sum of square differences (SSD) of the pixel values between boxes S_p and $\tau(S_p)$. In the proposed method, we have considered some features to represent the boxes instead of simple pixel values (RGB). The detail description of the features are mentioned in Section 3.2. The parameter σ is chosen manually to a high value. However, the choice is not critical for producing good output.

As we discussed, solution to the video inpainting problem would be given by the corresponding map τ that maximizes (2) defining self-similarity and coherency. Since the optimization problem (2) involves a non-linear objective function (NP hard), it is difficult to solve in a straight-forward method way. So we propose to proceed with iterative Expectation-Maximization (EM) algorithm. First, we initialize the correspondence map τ by random guess. In E-step, we generate target texture according to given correspondence map τ based on the source texture, and in the following M-step, we update the current guess of τ by assigning the boxes in the target region to the boxes in the source region. The E-step and M-step inherently maximizes the coherency and self-similarity of the filled region with the known portion respectively. We enforce self-similarity in the correspondence map τ by assigning nearby boxes by their neighbouring boxes. The coherence between boxes in T and those in rest of the video $V \setminus T$ as shown in eq. (2) is maximized if for every box $S_p \in \Delta$ all the surrounding boxes $[S_{p_1}, S_{p_2}, \ldots, S_{p_k}]$ agree on the box assignment at $\tau(S_p))$ with all the corresponding location of $[\tau(S_{p_1}), \tau(S_{p_2}), \ldots, \tau(S_{p_k})]$ appear in the video $V \setminus T$. Therefore, the iterative E-step aims to satisfy this condition for every box $S_p \in \Delta$, and the M-step searches for the best similar box in the box subspace $\Gamma \setminus \Delta$ of the unaltered region $V \setminus T$ of the video V. Let $[S_{q_1}, S_{q_2}, \ldots, S_{q_k}]$ denotes the boxes in $V \setminus T$ that are most similar to $[S_{p_1}, S_{p_2}, \ldots, S_{p_k}]$. Then the predicted S_{p_i} would be reliable if $s_i = s(S_{p_i}, S_{q_i}) \approx 1$. Therefore ,at each iteration, for each box $S_p \in \Delta$ and corresponding surrounding box S_{p_i}, we need to find out best possible box S_{q_i} in $V \setminus T$. Then we replace the box at S_p by the weighted average of the box at the corresponding locations of the similar boxes S_{q_i}. The weights are simply taken as the similarity measure s_i between the corresponding boxes S_{p_i} and S_{q_i}. Now the required huge computation for searching process to find the nearest neighbour (most similar) is reduced significantly by compensating it to approximate nearest neighbourhood [13]. This procedure may be expressed explicitly in the following way

E-step:

$$\widehat{V} := V \setminus T \cup \widehat{T} \tag{5}$$

where $\widehat{T}$ is obtained from T by replacing each 3D-box S_p by $\tau(S_p)$. Usually the boxes are overlapping and the boxes are aggregated in overlapping region. The aggregation is done by the weighted average of the overlapping boxes where the weights come from the similarity measure (SSD) between boxes S_p and $\tau(S_p)$.

M-step:

$$\begin{aligned} \widehat{\tau} &:= \arg\max_{\tau} \textstyle\sum_{S_p \in \Delta} s\left(S_p, \tau(S_p)\right) \\ &:= \arg\min_{\tau} \exp\left(\frac{-d_{SSD}\left(S_p,\, \tau(S_p)\right)}{\sigma^2}\right) \end{aligned} \tag{6}$$

where $\widehat{\tau}$ is the modified estimation of τ. At each iteration we update the target region T and for each box $S_p \in \Delta$ (full or partially unknown box space), we find most similar boxes (candidates) $\tau(S_p)$ as the box $S_q \in \Gamma \setminus \Delta$. If there are several boxes minimizing this quantity we select one arbitrarily. The iterative process should end when the correspondence map $\widehat{\tau}$ remains unchanged in two consecutive iterations, i.e., $\tau(S_p)$ assigns same $S_q \in \Gamma \setminus \Delta$, $\forall S_p \in \Delta$ in two consecutive iterations. However, the solution of the maximization problem may not converge to the actual global minimum or to a stationary point. As there is no guarantee that the iterative process converge to a stationary point, we set the stopping criterion as the maximum number of iterations.

3 Implementation Details

The algorithm is applied on a video V with target region T where inside T information is missing and the algorithm is supposed to infer missing information from the rest of the video $V \setminus T$. The proposed algorithm is an iterative method and in each iteration 3D boxes within T are refined according to the maximization of coherency and self-similarity.

Nearest neighbour search is an important problem in a variety of applications, including the knowledge discovery and data mining, pattern recognition and classification. The important task is to build a fast algorithm that could able to find nearest neighbour of every patches of an image. Obviously the problem can be solved in $O(dn)$ time through simple brute-force search for n points in d dimensional space. We use an efficient optimal algorithm for approximate nearest neighbour search [14] that can solve the problem in $O(d \log n)$ or less time.

Instead of using discrete patches, we use overlapping ones and use simple weighted average over the overlapping portions where the weights are computed as proportional to the similarity measure of the patches.

3.1 Multiscale Implementation

To enforce the global consistency further and also to speed up convergence, we perform the iterative process in multiple scales in both spatial and temporal directions. Each of the scale makes the resolution a fraction of the resolution of the upper scale. Scaling factor $1.25 - 2.00$ can produce significant result in most of the cases. In all of our experiments, we have chosen resolution scale to be 1.5 in both the directions. The optimization is done by using EM technique starting at coarsest scale and the solution is propagated to finer levels for further refinement. Initially, we fill the the target region for each frame of the video by some random texture at the coarsest scale followed by a few EM iterations. The filled region gets refined as iteration goes. Then, both T and $V \setminus T$ are gradually upsampled to finer resolutions, followed by more EM iterations, until the final fine resolution is obtained. We fix the number of iteration as the stopping criterion to terminate the process. However, we can terminate during the iterative process at the topmost scale when we achieve acceptable visual quality.

3.2 Spatio-temporal Similarity Measure

The Sum of Square Differences (SSD) of color channels is widely used in image inpainting but it is inadequate to produce the desired result in spatio-temporal space. Since human visual system is very sensitive to motion, a well-behaved measure needs to acknowledge human visual *perception*. For this we would like to incorporate some motion information into our algorithm. We add optical flow in our measure to obtain motion information. Suppose a pixel at location $p = (x, y, t)$ in one frame moved to $(x + \partial x, y + \partial y, t + \partial t)$ in the next frame. Then $v_x(p) = \partial x / \partial t$ and $v_y(p) = \partial y / \partial t$ gives the motion estimation at p. If the motion is only in the horizontal direction, then u_x captures the instantaneous motion in the x direction. Similarly for vertical direction, v_y captures instantaneous motion in the y direction. These two measures depend upon the spatial and temporal changes while capturing object velocities. We add these two components after scaling the RGB values to obtain a five-dimensional representation for each space-time point: (R, G, B, v_x, v_y). We apply SSD to this $5D$ feature vector to capture spatial and temporal similarities simultaneously. So for two space-time boxes S_p and S_q, we have $d_{SSD}(S_p, S_q) = \|u(S_p) - u(S_q)\|_2^2$. We take nearest neighbours of each box in the target portion T using the distance measure d_{SSD} on the 5D representation of space-time points and keep it to update the pixels in the target region.

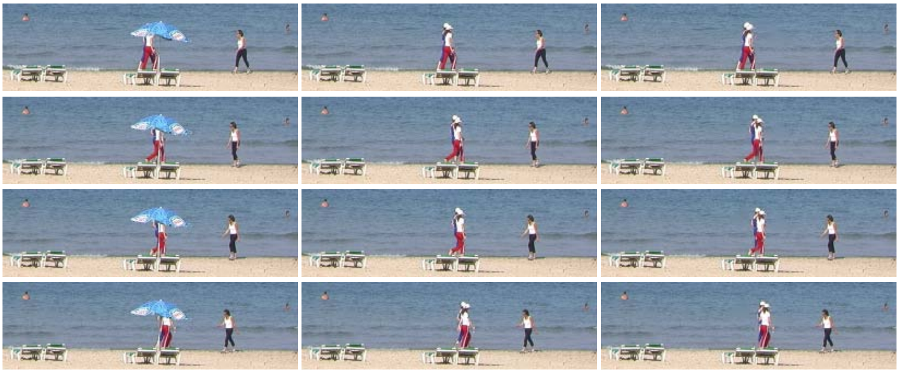

Fig. 1. Results of proposed video inpainting method. The first column contains some frames of the input video where the umbrella is removed during video completion. The second column are the result of Wexler *et al.* [15] and the right column represents resultant frames using proposed method.

4 Experiment and Results

In this section experimentally we set different parameters used in our algorithm and test it for some videos. In our experiment, the size of the box is chosen as 5 × 5 × 5 with 3-pixel overlap. For faster synthesizing process, the number of scale in the pyramid is chosen as 4 with resolution factor 1.6 and the number of iterations in each scale is chosen as 10.

In Fig. 1, we displayed the result of proposed video completion techniques. We have displayed only some frames to demonstrate the motion preservation of the object while filling the target hole. We observe comparable result with Wexler *et al.* [15] and no additional artificial distortions occurred during box-based filling. This result is generated within 20 minutes where as Wexler *et al.* had reported more than 10 hrs to get that output. with other videos also we experience these faster completion.

5 Conclusion

In this paper we describe a fast video inpainting algorithm by maximizing box-based self-similarity and coherency in a comprehensive framework. We combine these two concepts into a maximization problem and optimize by EM-algorithm to produce the inpainted video. Experimental results show that the output of the proposed method is comparable to existing approaches.

References

1. Bertalmio, M., Sapiro, G.: Image In painting. In: Proc. of the ACM SIGGRAPH Conf. on Computer Graphics, New York, USA, pp. 417–424 (2000)
2. Efros, A.A., Freeman, W.T.: Image quilting for texture synthesis and transfer. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, pp. 341–346. ACM, New York (2001)
3. Criminisi, A., Perez, P., Toyama, K.: Object Removal by Exemplar-Based Image Inpainting. In: Proc. IEEE Int. Conf. on Computer Vision, vol. 2, pp. 721–728 (2003)
4. Sun, J., Yuan, L., Jia, J., Shum, H.: Image Completion with Structure Propagation. In: Proc. SIGGRAPH 2005, pp. 861–868 (2005)
5. Ashikhmin, M.: Synthesizing natural textures. In: Proc. of ACM Symp. on Interactive 3D Graphics, pp. 217–226 (2001)
6. Komodakis, N., Tziritas, G.: Image Completion using global optimization. In: Proc. IEEE Computer Soc. Conf. Computer Vision and Pattern Recognition, pp. 442–452 (2006)
7. Bugeau, A., Bertalmio, M., Caselles, V.: A comprehensive framework for image inpainting. IEEE Transaction on Image Processing 19 (October 2010)
8. Wexler, Y., Shechtman, E., Irani, M.: Space-time video completion. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. I.120–I.127 (2004)
9. Jia, J., Pang Wu, T., Wing Tai, Y., Keung Tang, C.: Video repairing: Inference of foreground and background under severe occlusion. In: Proc. Computer Vision and Pattern Recognition, pp. 364–371 (2004)
10. Sapiro, G., Patwardhan, K.A., Bertalmio, M.: Video inpainting of occluding and occluded objects. In: IEEE Int. Conf. Image Processing., vol. 2, pp. 69–72 (2005)
11. Patwardhan, K.A., Sapiro, G., Bertalmio, M.: Video inpainting under constrained camera motion. IEEE Transc. on Image Processing 16, 545–553 (2007)
12. Efros, A.A., Leung, T.: Texture synthesis by non-parametric sampling. In: IEEE International Conference on Computer Vision (ICCV), vol. 2, pp. 1033–1038 (1999)
13. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: Patchmatch: a randomized correspondence algorithm for structural image editing. ACM Transactions on Graphics, SIGGRAPH, 24.1–24.11 (2009)
14. Arya, S., Mount, D.M., Netanyahu, N.S., Silverman, R., Wu, A.Y.: An optimal algorithm for approximate nearest neighbor searching fixed dimensions. Journal of ACM 45, 891–923 (1998)
15. Wexler, Y., Shechtman, E., Irani, M.: Space-time completion of video. IEEE Transactions on Pattern Analysis and Machine Intelligence 29, 463–476 (2007)