EXEMPLAR-BASED IMAGE COMPLETION BY COMBINING INTERACTION AND PATCH OFFSETS

Weiwei Xue, Bodong Zhang, Rong Zhang

Department of Electronic Engineering and Information Science,
University of Science and Technology of China, Hefei, China

ABSTRACT

Image completion solves the problem of filling missing regions in a visually plausible way with consistent textures and continuous structures. Completion of complicated structures and low-efficiency problem are two main challenges for exemplar-based approach. In this paper we take advantage of sparsely distributed patch offsets (relative positions) and its representativeness for the main structures, which can be used to decrease the search scope of similar matching without sacrificing completion quality. We further put forward creatively converting simple user interaction to a set of patch offsets and unified the two kind of offsets to form a concise but comprehensive candidate collection. Experiments on various types of images including challenging cases show the robustness and effectiveness of our method.

Index Terms— Image completion, interaction, exemplar-based, patch offset.

1. INTRODUCTION

Imagine you are taking photos of a fleeting scenery, an unexpected visitor burst in on the moment you hit the shutter button. You may feel annoyed until you find a magic tool which can remove the unexpected object from the photo. This is a typical usage scenario of image completion. Generally speaking, image completion focuses on filling the missing parts in images by using the information of the same or another image, which is an important topic in the field of image processing and computer vision. Generating visually plausible result, which preserves the continuity and consistency of the structure and texture information, is the ultimate goal of image completion.

One category of image completion methods is diffusion-based, which appears in the early stage of this technique. These methods regard image completion as solving partial differential equations (PDEs) or variation problems by using the known pixels around the missing regions as boundary conditions [1, 2]. This kind of methods generates pixels in unknown regions approximately under the assumption of color smoothness and continuity on the image. They work well for the situation of narrow scratches or small holes, but produce blurring artifacts for large missing regions due to the missing of structure information.

Another category is exemplar-based methods, which regards the known regions as texture patches and performs example-based texture synthesis method to generate the new information for the missing regions [3-7]. Criminisi et al. [5] proposed an algorithm considering the importance of the structural information, which makes the missing regions with strong structures have higher priorities in completion process. Good completion quality makes it a milestone. However, it's difficult to maintain the structural consistency when meeting with complicated structures. Sun et al. [8] raised an algorithm using interactive method to guide the result by providing hints for the computer, which overcome this problem. PatchMatch [9] framework is an user-guide method to finish diverse tasks including image completion, which is a high-level interactive image editing tool and also represents the stat-of-art level of this technology. However, inefficiency issue always makes exemplar-based methods be far from practical application.

Two essential properties for a good completion algorithm can be summarized from development of image completion: (1) Complicated structures must be completed in a visual plausible way as well as textures, for human eyes are more sensitive to inconsistent of structures. (2) The algorithm must be efficient enough that users can immediately get the result. Users can use the tool to try and create the desired result.

Besides, user interaction is essential for image completion, for the following reasons: (1) First, user interaction provides important information that is known to him but not to the computer, such as significant structure line and reserved region, which can obtain the best result. (2) Second, users often cannot articulate his completion goals a priori, so interaction gives him an efficient way towards a satisfactory result by the process of trial and error.

To overcome the drawback of exemplar-based methods and take advantage of interaction, we propose a novel and effective approach using patch offsets and interaction in this paper. On one hand, we are inspired by the observation that the statistics of patch offsets are sparsely distributed among natural images [10], the number of similar candidates will be greatly reduced using this statistic feature. On the other

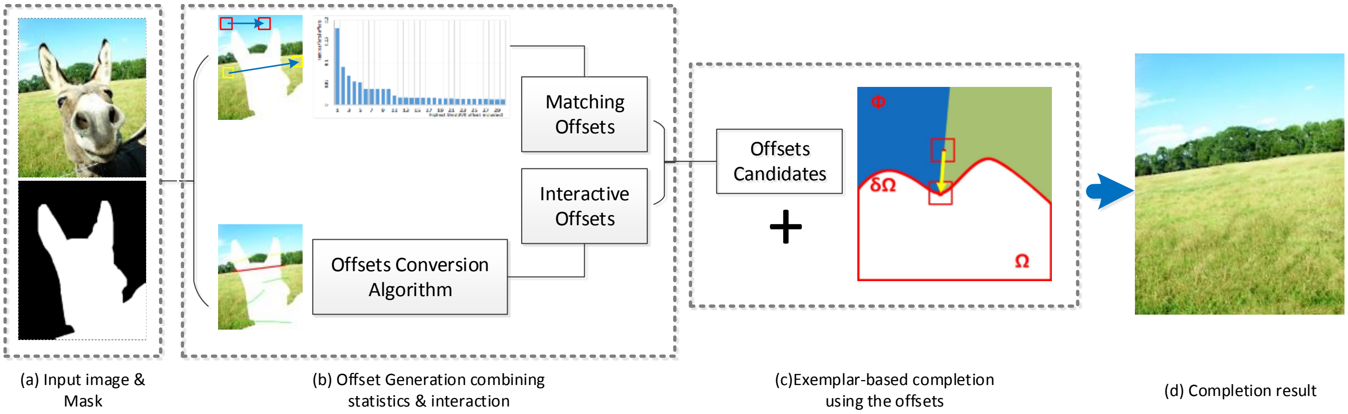


Figure 1 Outline of our approach. Image was taken from MSRA Salient Database [11].

hand, we convert user interaction to significant offsets and add it to statistical offsets to form a unified and complete candidate collection. Experiment results show our method can observably enhance the visual quality of the completion images and better than other exemplar-based results.

2. APPROACH

In this section we will first introduce the significant role patch offsets played in promoting exemplar-based completion algorithm. Then we will introduce our creative shining point of converting user interaction to patch offsets. The outline of our approach is shown in Figure 1.

2.1. Matching by Statistics of Representative Offsets

As we know, the classical exemplar-based algorithm raised by Criminisi [5] can be formulized as

$$\Psi_{\hat{q}} = \arg\min_{\Psi_{\hat{q}} \in \Phi} d(\Psi_{\hat{p}}, \Psi_q) \qquad (1)$$

Here, $d(\cdot,\cdot)$ means the differences of two similar patches. The formula means if we have a patch Ψ_q in the boundary of unknown region and find it's best matched patch $\Psi_{\hat{q}}$ in the known parts using the know part of Ψ_q, then unknown pixels in unknown parts of Ψ_q will copy from $\Psi_{\hat{q}}$ to finish the completion. The coordinate difference between Ψ_q and $\Psi_{\hat{q}}$ is called offset of Ψ_q.

We noticed that the bottleneck of the whole process is the step of matching, the original algorithm search in the whole image to find the most similar patch for every block in the missing region. There are many researchers try to solve this low-efficiency issue from decreasing the search scope to designing a new data structure for acceleration. But these methods sacrifice the result quality or waste storage space, which are not fundamental solutions. We start from a distinct perspective by thinking about there must exist some features standing for the whole image, which is similar to the form of sparse representation.

We are inspired by the observation that the statistics of patch offsets are sparsely distributed among natural images and the prominent offsets contain the main structures of the image [10], which means a few dominant offsets can represent the whole image well (shown in Figure 2). Hence, on one hand, using these prominent offsets will decrease the search scope of similar patch matching and accelerate the speed of this algorithm. On the other hand, as we know human eyes are more sensitive to structural line than flat textures in images, more coherent completion result will be gotten for this offsets including repetitive structures and patterns in the image. Here we show a simple algorithm transferring the repetitive patterns to offsets and make use of the prominent offsets to accelerate the algorithm.

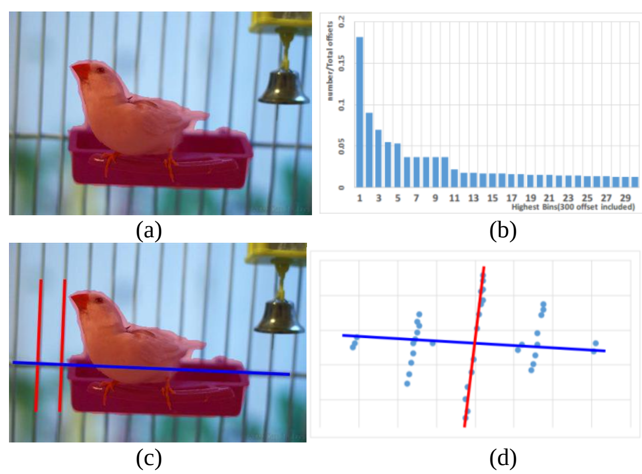


Figure 2 Sparsely distribution of offsets. (a) original image with red mask. (b) If we calculate the histogram of the offsets, we find most of the offsets locate in few number of bins, which will form a long-tail phenomenon. (c) & (d) Prominent offsets can represents the main structure lines in the image.

Finding Nearest Neighbor Fields. In order to get the prominent offsets, we do statistics for similar patch matching, which we call nearest neighbor fields (NNF). For every patch P in source region we find another known patch that is the most similar with P and compute their relative position, which we call offsets. Defined as formula,

$$\tau(\mathbf{x}) = \arg\min_{\tau} \parallel P(\mathbf{x}) - P(\mathbf{x}+\tau) \parallel^2, \mid \tau \mid > \theta \qquad (2)$$

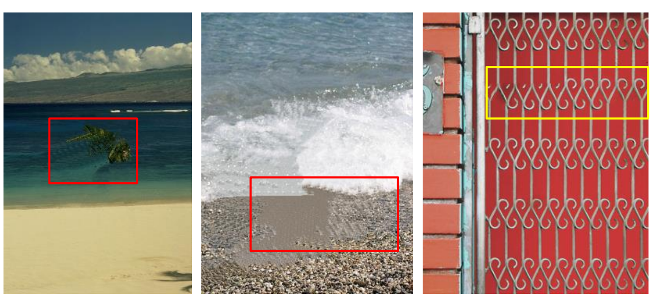
Here, $\mathbf{x} = (x, y)$ is the position of a $w \times w$ patch centered at $\mathbf{x}$, $\tau = (u, v)$ is the 2-d coordinates of the offset which stand for the distance vector of the patch and its most similar one. The distance is defined using the sum of squared differences (SSD) between the two patches as smaller SSD value corresponds to higher similarity. Especially, the threshold θ is used to ignore the nearby patches which are likely to be similar but do not contribute to the repetition offset.

Offsets Generation. After extracting all the offsets from all known patches, we generate the histogram of all the offsets. Assume that the offset τ stand for a point in 2-d plane, we calculate the histogram as,

$$H(u, v) = \sum_{\tau(\mathbf{x})} \delta(\tau(\mathbf{x}) = (u, v)) \qquad (3)$$

Here $\delta(\cdot)$ equals to 1 if the argument is true and 0 otherwise. $H(\tau)$ gives the amount of patches having offset as τ, to reduce the influence of noise and make the histogram smoother, we smooth the histogram using a Gaussian filter. And then we pick out the K highest peaks of the histogram, we define peak as a bin whose magnitude is locally maximal. In our experiment, we set $K = 1000$.

Exemplar-based completion. We use selected prominent offsets to limit the size of candidate collection rather than searching whole known pixels of traditional methods. Simply exhausting $i \in [1, K]$ for each $P(\mathbf{x})$ to find the most similar one $P(\mathbf{x} + s_i)$, we can run the algorithm more efficiently and get better result than the other methods.



(a) Too Large Area (b) Random Textures (c) Complex Stucture

Figure 3 Some situation that cannot be treated well just using the patch offsets. (a) The missing area is too large for completion, which means offsets collection cannot cover the offset needed by the middle pixels. (b) Offsets cannot cover the random situation of random textures. (c) Complex structure need more effective offsets to be completed.

2.2. Convert User Interaction to Patch Offsets

The utilization of prominent patch offsets will get visually plausible results in a very short time, which can meet the requirement of real-time image processing. But there are still some situation that cannot be treated well, such as, a) the size of missing hole is too large that beyond the offsets collection. b) There may be artifacts for random textures when the offsets quantity is insufficient. c) Automatic completion of very complex structure is sometimes difficult. (as Figure 3 shows)

The additional information provided by the users' interaction opens a door for solving the above problems. Impressive guided filling results were shown by annotating structures that cross both inside and outside the missing region [8]. The uniform framework for image editing proposed by Barnes et al. [9] proved that interaction played significant role in promoting image editing quality.

We proposed a simple but smart algorithm for converting interaction path to offsets with different sizes, which will be effective supplement of automatically generated offsets. The combination of the two parts offsets form a brief but comprehensive candidate for exemplar-based image completion.

Quantization. We convert user interaction to offsets by using uniform quantization, which means one vector can convert to several vectors with different multiple. See Figure 4 as a example (straight line), here vector V_0 stand for the interaction line supplied by user. We divide V_0 for n parts, and then we can get n vectors with different length in the same direction. Formally, we convert interaction line V_0 to a set of offsets: $C_i = \{V_1, V_2, V_3, ...V_n\}$. And for curve line, we evenly divided the length of arc, and the take the vector from start point to every divided point for offset. We enlarge the n value to make the quantization more smooth for curve lines.

Combine with statistics. We then combine the offset from statistics and interaction together to form a larger collection for completion. As the formula shows,

$$C_t = C_a \cup C_i \qquad (4)$$

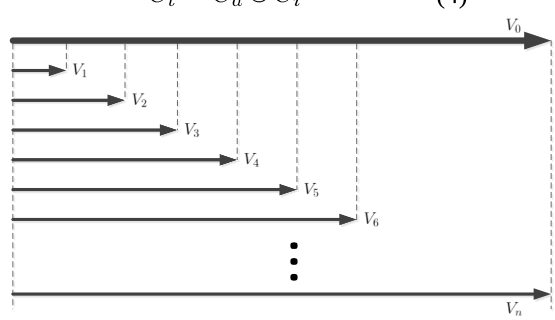


Figure 4 Quantization method converting user's interaction to multiple vectors.

3. EXPERIMENT RESULTS

3.1. Important Role of Interaction

As we have mentioned in Section 2.2, user interaction will overcome the drawback that statistics cannot provide sufficient offsets for some typical case. We can see from Figure 5 that visually plausible result will be get by combining user interaction for the failing case in Section 2.2. Interaction provides offsets across the known and missing region to make sure all patch in missing region can be covered by offsets and also provide more sufficient offsets both for random texture case or complicated structure cases.

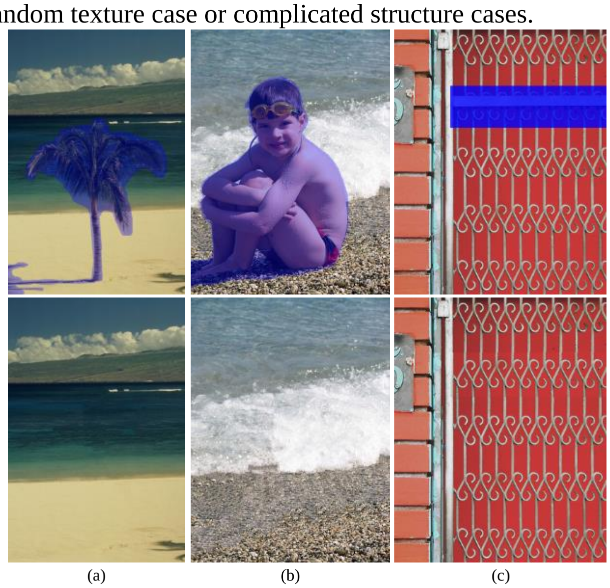


Figure 5 Combining with user interaction will create more visually plausible result than just using statistical offsets.

Further more, interaction can also give user the chance to continuously trial forward a better result for the reason that they may not articulate completion result a priori. Here is an example of using different interaction to give different but both visually plausible completion result (Figure 6).

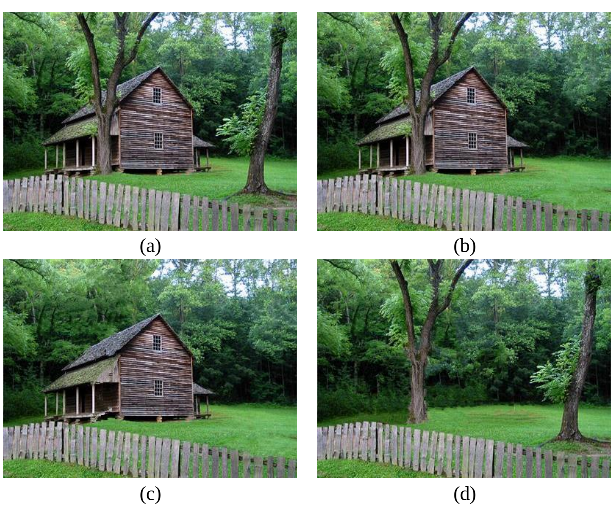


Figure 6 Interaction can be used for designing different but plausible results. (a) Input. (b) Complete the right "tree". (c) Complete the left "tree". (d) Keep the two trees and complete the "house".

3.2. Comparison with Other Methods

We compare our approach with other well-known exemplar-based methods including the traditional one [5], the interactive method with good result [8] and the state-of-art method using by Adobe Photoshop (Content-Aware Fill Tool) [9]. The results are shown in Figure 7. Criminisi's method [5] cause error accumulation effect when images contain complexity structures (see Figure 7(c)), Sun's method [8] complete structures with a simple curve provided by user, but we can still find small visual incoherence from the result (Figure 7(d)). [9] is a well-tuned and optimized method which produce high quality result in many practices, we get a neck and neck result as their method but from a simpler perspective and obviously more efficient (Figure 7(e&f)).

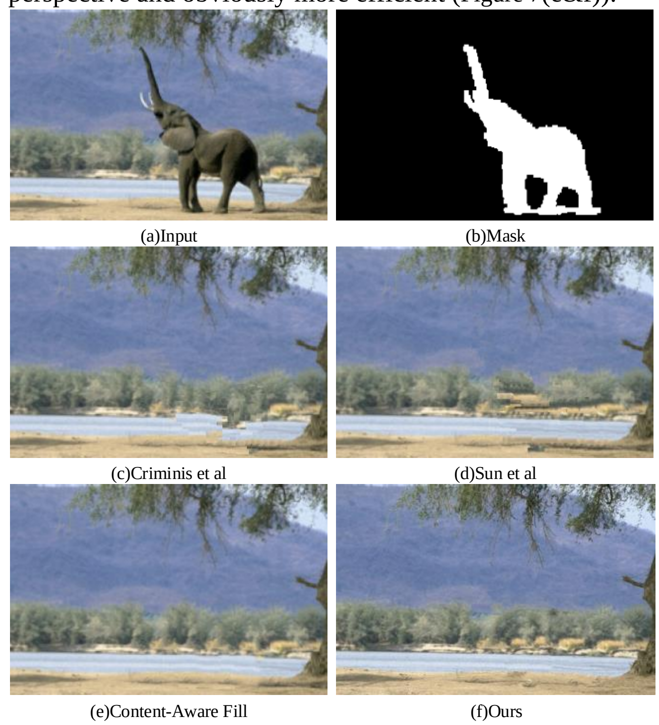


Figure 7 Comparison with other methods

4. CONCULUSION

In this paper, a novel exemplar-based image completion algorithm combining patch offsets and user interaction is proposed. We show the superiority and effectiveness of using representative offsets rather than all the patches in the known region. And the conversion from user interaction lines to offsets has also been proved to provide a sufficient supplement of information for image completion, which lead to a better result than just using the statistical offsets. Experiment results on various types of images show the advantages of proposed algorithm in overcoming the drawbacks of other popular techniques.

In the future, we will further investigate higher dimensional transformation spaces like rotation, scaling, reflection for generating more plentiful kinds of offsets.

5. REFERENCES

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image Inpainting," *Proceedings of SIGGRAPH 2000*, 2000.

[2] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, "Simultaneous structure and texture image inpainting," *Image Processing, IEEE Transactions on*, vol. 12, pp. 882-889, 2003.

[3] T. H. Kwok, H. Sheung, and C. C. L. Wang, "Fast query for exemplar-based image completion," *Image Processing, IEEE Transactions on*, vol. 19, pp. 3106-3115, 2010.

[4] F. Tang, Y. Ying, J. Wang, and Q. Peng, "A novel texture synthesis based algorithm for object removal in photographs," *Advances in Computer Science-ASIAN 2004. Higher-Level Decision Making*, pp. 3299-3300, 2005.

[5] A. Criminisi, P. P ́erez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *Image Processing, IEEE Transactions on*, vol. 13, pp. 1200-1212, 2004.

[6] W. H. Cheng, C. W. Hsieh, S. K. Lin, C. W. Wang, and J. L. Wu, "Robust algorithm for exemplar-based image inpainting," in *International Conference on Computer Graphics, Imaging and Visualization*, 2005.

[7] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, 1999, pp. 1033-1038.

[8] J. Sun, L. Yuan, J. Jia, and H. Y. Shum, "Image completion with structure propagation," *ACM Transactions on Graphics (ToG)*, vol. 24, pp. 861-868, 2005.

[9] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "PatchMatch: a randomized correspondence algorithm for structural image editing," 2009, p. 24.

[10] K. He and J. Sun, "Statistics of Patch Offsets for Image Completion," *ECCV*, 2012.

[11] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, 2007, pp. 1-8.