

# Effective structure restoration for image completion using internet resources

Miao Hua<sup>1,2</sup> · Wencheng Wang<sup>1</sup>

Published online: 9 May 2015  
© Springer-Verlag Berlin Heidelberg 2015

**Abstract** Structure restoration plays an important role in image completion. Though much progress has been made, existing techniques are ineffective to restore plausible complex structures, due to the high complexity of structure registration between neighboring regions. We address this challenge by taking two measures. The first is to get the images whose structures are potential to be well merged with the structures around the hole. This can be achieved by effectively retrieving the mass images on the internet, which provide enough candidates. The second is to coherently blend the retrieved structures with the structures around the hole, which is by suppressing the connection differences between these structures. Experimental results show that we can easily complete images, especially with complex structures or for filling large holes.

**Keywords** Image completion · Structure retrieval · Iterative closest point

## 1 Introduction

Image completion, also known as image inpainting or hole filling, is the task of filling or replacing an image region (called a *hole* in the following description) with new image

data. Here, the challenge is to make the filled image content blended with the original image coherently. With regard to this, structure information plays an important role, because human vision is liable to ignore large uniform area and sensitive to structural regions, as pointed out by Nill and Bouzas [24]. To coherently restore the structures, many techniques have been proposed. Some use user interaction to specify the structure in the hole, and some try to automatically propagate structure information from the source image into the hole. Though they have made much progress, they are not effective to fill large holes or fill holes with complex structures. These are discussed in the following paragraphs.

With regard to the approaches via user interaction, they allow the user to draw strokes to specify structure information in the hole [30], or sketch the desired object shape in the hole [13], so that the user's visual experiences can be taken to guide structure restoration. However, the user's visual experiences are limited, preventing using complex structures. Moreover, in filling a large hole, it may be troublesome and very laborious for the user to specify the structures.

In a lot of methods, structure restoration is implicitly handled in using image patches to fill the hole, such as having each patch in the filled region as similar as possible to a known patch [2, 28, 31, 32], or having the neighboring patches/pixels in the filled region from visually coherent content [15, 20, 27]. They are based on the observation that structure information is much correlated with the neighboring information. When neighboring regions are consistent, the related structures can be restored correspondingly. However, in treating complex structures or filling large holes, such a correlation is not reliable. This is because complex structures will have much complicated correlation, and the large holes will increase the complexity to use such correlation, both of which are very possible to cause wrong structure reg-

---

✉ Wencheng Wang  
whn@ios.ac.cn

Miao Hua  
huam@ios.ac.cn

<sup>1</sup> State Key Laboratory of Computer Science,  
Institute of Software, Chinese Academy of Sciences,  
Beijing 100190, China

<sup>2</sup> University of Chinese Academy of Sciences,  
Beijing, China

istration between neighboring regions, leading to artifacts in structure restoration.

There are also many methods explicitly using structure information for image completion [3,8,19], but they are not effective to complete images with complex structures or fill large holes either. Recently, a method [18] suggested to propagate structure information explicitly to guide image completion. In this method, salient curves are detected to represent structure information, and the curves from different image pieces are registered to restore structures in the gaps between these image pieces. When the gaps are not large and the structures of the image pieces are not complicated, this method works very well because in such a case it is easy to register these curves. However, it cannot well treat complicated structure restoration due to the registration difficulty, listed as a limitation in this paper. Similarly, it cannot well restore structures in large gaps.

In this paper, we address this challenge of effectively restoring structures for plausible image completion. Our solution is to reduce the structure registration complexity by getting suitable structures that can be connected well with the structures around the hole, and then blend these structures coherently by suppressing the differences between them. For getting suitable structures, we try to retrieve the mass images on the internet, which provide enough human visual experiences on structures. This is motivated by the study that people always make predictions about what may exist in the world beyond the image frame using visual associations or context [1]. In blending the retrieved structures with the structures around the hole, we design a new energy function based on Iterative Closest Point (ICP) [4,5] to progressively refine the propagation of the retrieved structures in the hole for consistent connection, which is difficult to obtain with existing techniques. After the structures are restored in the hole, they are used to guide image completion. Figure 1 shows the framework of our system for image completion.

In sum, we have specific contributions in the following:

- A novel image completion approach that can easily treat large holes and use complex structures.

- A data-based system for image completion using internet resources, with which the user only needs to select the reference image from the retrieved ones, saving much manual labor.
- A new energy function to effectively propagate retrieved structures for coherent restoration in the hole, superior to existing techniques.

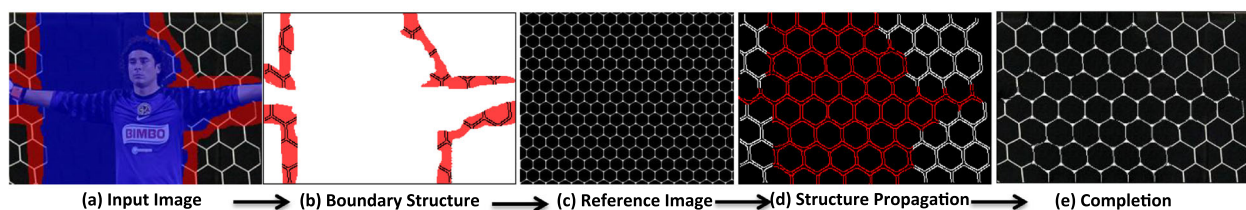
## 2 Related work

### 2.1 Image completion

The methods for image completion have been proposed in a large quantity. By the manner for structure restoration, they can be classified into two categories, the one in an explicit manner, and the other in an implicit manner.

For the methods in an explicit manner, some tried to infer plausible structures in the hole. For example, Bertalmio et al. [3] decomposed the image into the sum of two functions, corresponding to the structure information and the texture information, respectively, which are used to fill the hole, respectively, in two sub-images that are combined afterwards. Jia and Tang [19] took a texture-based segmentation manner to translate image color and texture information into an adaptive ND tensor, followed by a voting process for image completion. Criminisi et al. [8] suggested the order in which the filling is proceeded, to accord with structure propagation. Other work tried to get information from stereoscopic images [23]. Recently, Huange et al. [18] extracted salient curves as structures and used a tele-registration method to align the image pieces to fill the gaps between those pieces. There are also many methods proposed to manually specify structures in the hole, using strokes [30] or sketches [13]. Clearly, in treating large holes or using complex structures, such a way is not efficient and very difficult to restore plausible structures, because of the high complexity of coherent structure connection between neighboring regions.

For the methods in an implicit manner, they are generally based on the observation that when neighboring regions are consistent, the related structures are restored correspond-



**Fig. 1** Framework of our system. Given an input image (a), where the hole is marked in blue, and the surrounding region in red. The structures in the surroundings are extracted (b). By the structures in the surroundings, the reference image with matchable structures is retrieved

(c). Then, the retrieved structure is propagated in the hole and blended seamlessly with the structures in the surrounding region (d). Finally, the image is completed guided by the propagated structures (e)

ingly. Thus, Wexler et al. [31,32] optimized a global cost function to have each patch in the filled region as similar as possible to a certain known patch. To speed up the optimization process, a fast PatchMatch algorithm was proposed [2]. Other methods optimized graph-based models like Markov Random Fields (MRFs), as in the works of Priority-BP [20] and Shift-map [27]. In [9,22], structure information under a variety of transformations such as scale, rotation and brightness change was also considered for quality image completion. To effectively employ reliable information, it was proposed to use a few dominant patch offsets [15]. Though much progress has been made, they are difficult to restore quality structures in the hole in many cases, especially in treating large holes, because limited candidates increase the difficult to get suitable structure registration between neighboring regions.

Our work follows the framework of the work by Hays and Efros [14], the first method to complete images with image retrieval. It found the similar scenes which contained image fragments that could convincingly complete the input image, and then used them to fill the holes by minimizing the gradient of the image difference along the seams between them. Unfortunately, this method did not take into account structure restoration in image completion, and so may fail in restoring plausible structures in many cases. As for our work, we try to find the images with similar structures instead of similar scenes and then propagate structures to seamlessly blend with the structures around the hole, so that structures can be well restored for high-quality image completion.

## 2.2 Image retrieval

To take advantages of the abundant image resources on the internet, many works have been done [17], such as for image editing by semantic colorization [7] and object manipulation in images [13]. Here, an important task is to retrieve suitable images from the internet. For this, some methods used key words to search on the internet and applied a series of filters. In this aspect, the techniques via descriptors are very effective, such as the scene descriptors for finding similar scenes for scene completion [14]. As our method is based on structure information for image retrieval, and the structures are represented by the extracted edges, which are very like sketches, we adopt the sketch-based image retrieval method in [11] in our system. To have the images retrieved very effectively, we apply several measures to promote edge extraction and the SHoG descriptor, which is used in [11] for image retrieval.

## 2.3 Image blending

There are a lot of works on image blending for various applications, such as optimizing the affine transformation between

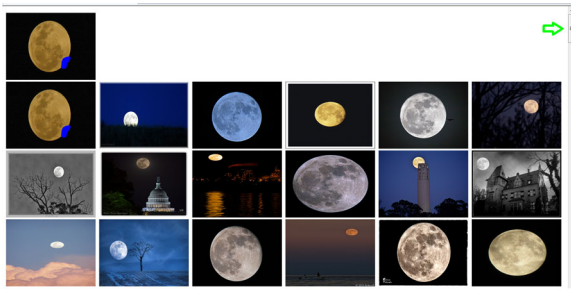
the structures in the reference image and the input image for consistent blending [6]. It is out of the scope of the paper to survey them. In our system, image blending is executed after the structures are restored, so that we can care less attention on the structure consistency in the blending process. Considering the efficiency, we mainly adopt the method in [15] for image blending, which takes patches from the reference image to fill the hole, and runs very fast. As this method may spend much time on optimizing the patch selection, we adopt Poisson cloning [25] for image blending when the reference image has similar appearance with the input image, because Poisson cloning can achieve a seamless blending by smoothing the differences between the input image and the reference image on the border of the hole across the reference image, saving the selection of patches.

## 3 Structure retrieval

Our structure retrieval is using the structure information in the surroundings of the hole. In [30], it is investigated that most salient structures can be approximated using a few well-defined curves, and these curves are composed of the extracted edges from the image. Thus, we extract edges in the surroundings of the hole as the query template for structure retrieval. For robust extraction, we first filter the input image using the edge-preserving WLS filter [12], which can sharpen major edges while eliminating low salient structures, then apply the technique in [18] to the filtered image for edge extraction. After the edges are extracted, they are regarded as sketches to retrieve images on the internet using the sketch-based image retrieval approach [11].

In implementing the retrieval approach, many things should be carefully considered. The first is the width of the surrounding region of an image hole. If the surrounding region is larger, more edges would be extracted, which may be useful to get more plausible structures with more constraints, and since there are more edges, fewer images with similar structures will be obtained in the internet. On the contrary, a smaller surrounding region will lead to fewer edges extracted, which may have the retrieved structures not plausible for image completion. Of course, larger surrounding regions will lead more cost on edge extraction and retrieval processing. Thus, as a balance, we set the width of the surrounding region of an image hole as  $0.4 \times |H|$ , where  $|H|$  is the diameter of the image hole, which works well in our tests.

The descriptor we adopt is the SHoG descriptor [11]. To generate the descriptor, we used  $4 \times 4$  spatial and eight orientational bins, as used in many methods. We then randomly sample 1 million local features and use them for learning a visual vocabulary using standard  $k$ -means clustering, which costs about 30 h. In our implementation,  $k = 1000$  clusters were used. For a query, we generate the SHoG descriptor



**Fig. 2** Our interface for structure retrieval. The *first row* shows the input image with a hole in *blue*, and the *other rows* show the retrieved results, which are ordered by retrieval scores. The *rightmost pull-down bar* denoted by a *green arrow* helps the user to survey the retrieved results

for the extracted edges around the hole, and quantize it to be words, with which the matched images are looked up from the dataset, and ranked using the tf-idf weighting as in [11,29].

To well exploit the image resources in the internet, we built a dataset covering 250 common object categories as in [10]. For each object category, we downloaded about 5000 images from the internet and made a collection of 1.25 million images.

Figure 2 shows our interface for displaying the retrieved images, among which the user chooses one or more for image completion. In general, we display the top 500 retrieved images, and resize these images each into  $200 \times 150$  pixels to display so that the user can watch more in the limited screen space.

## 4 Structure restoration

After the reference image with plausible structures is chosen by the user, we first localize the region that contains the structures which are able to match with the boundary structures, called the *reference structures*. Then, the reference structures are used for structure restoration.

For localizing the reference structures, we resort to the directional chamfer matching technique in [21]. It takes

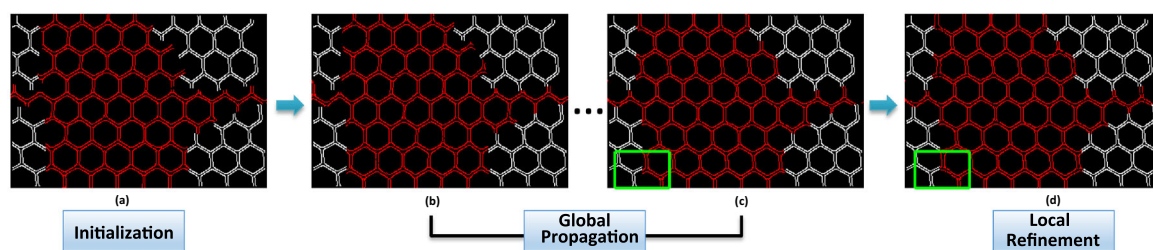
the boundary structures as the query template, where edge orientation information is considered in computing the chamfer distance to improve the matching accuracy.

In placing the reference structures for quality structure restoration, it should be carefully considered the differences between the reference structures and the boundary structures on scales, edge orientations, edge lengths and so on. With regard to this, we take a two-step algorithm, as illustrated in Fig. 3, where a global optimization process is first used to adjust the reference structures as a whole, and then the unmatched edges of the reference structures are adjusted locally to consistently connect with their corresponding edges in the boundary structures. These two steps are discussed in the following two subsections.

### 4.1 Global optimization

In global optimization, we first initialize the placement of the reference structures using the shape registration method in [16], which estimates the differences between the reference structures and the boundary structure to derive an affine transformation for aligning the reference structures with boundary structures. Then, the placement is iteratively optimized using an Iterative Closest Point (ICP)-based method [4,5]. ICP is an algorithm often employed to minimize the difference between two clouds of points. Here, we use the algorithm to iteratively revise the transformation (translation and rotation) to reduce the differences between reference structures and boundary structures on alignment.

In the alignment, it is very important to have the endpoints of the edges in the boundary structures, called *docking points*, connected consistently with the endpoints of the edges in the reference structures, called *anchoring points*, as illustrated in Fig. 4. Clearly, docking points and anchoring points can be extracted automatically, and for each docking point, its anchoring point is the nearest point to it in the reference structures. However, docking points and anchoring points may not be matched very well in general. Thus, we apply affine transformations to deform the reference structures to improve connection of these endpoints. If the alignment after

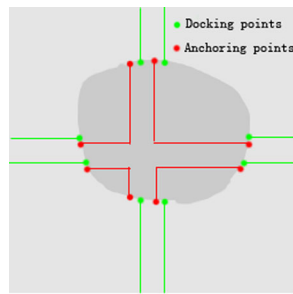


**Fig. 3** Structure restoration in two steps. After the reference structures are initially placed inside the hole (a), the placement of the structures is optimized as a whole iteratively (b) and (c). Afterwards, local refine-

ment is taken to well align the reference structures with the boundary structures, as marked in the *green windows*



**Fig. 4** Illustration of docking points and anchoring points, where the image hole is in gray



affine transformations is still not good, we iteratively apply affine transformations to further deform the reference structures for better alignment. Formally, this is by minimizing the following energy function:

$$\sum_{q_i \in Q} \min_{L, p_j \in P} w_i (\|L(p_j) - q_i\|^2 + \lambda \|L(\phi(p_j)) - \phi(q_i)\|^2) \tag{1}$$

with  $w_i = \min_{p_k \in P} \|p_k - q_i\|$  where  $Q$  is the set of the docking points,  $P$  is the set of the anchoring points,  $\lambda$  is a weight and set to 1 in all our experiments,  $L(x)$  is the affine transformation consisting of a linear transformation matrix  $M$  and a translation  $T$ , represented as  $L(x) = Mx + T$ ,  $\phi(x)$  is the edge orientation at point  $x$ ,  $L(x)$  and  $L(\phi(x))$  represent the position and edge orientation at pixel  $x$  after the affine transformation is applied to all points on the edges in the reference structures.

Let  $SP$  be the set of all points on edges of reference structures, we list the pseudo-code of our optimization algorithm in Algorithm 1. In line 3 of Algorithm 1, we exhaustively examine every anchoring point in  $P$  to find the nearest point for each docking point, and this strategy is relatively fast since the set  $P$  is generally small. In each iteration, the transformation  $L$  is derived using the standard least square method.

---

**Algorithm 1** GlobalOptimization()

---

**Require:** Docking points set  $Q$  and edge points set  $SP$  of the reference structures

**Ensure:** Global optimized  $SP$

- 1: Initialize  $P$
  - 2: **for** iteration  $n = 0 \rightarrow N$  **do**
  - 3:  $p_i = \arg \min_{p_k \in P} \|p_k - q_i\|$ , for any  $q_i \in Q$
  - 4:  $L = \arg \min_L \sum_{q_i \in Q} w_i (\|L(p_i) - q_i\|^2 + \lambda \|L(\phi(p_i)) - \phi(q_i)\|^2)$
  - 5: **if**  $p = L(p)$  for any point  $p \in SP$  **then**
  - 6:     Return  $SP$
  - 7: **else**
  - 8:      $p = L(p)$  for any point  $p \in SP$
  - 9: **end if**
  - 10: recompute  $P$
  - 11: **end for**
  - 12: Return  $SP$
- 

## 4.2 Local refinement

With global optimization, the reference structures can be placed well in the image hole as a whole, but it cannot be guaranteed that every anchoring point is matched well with a docking point, which may lead to serious artifacts. As illustrated in the green window in Fig. 3c, some anchoring points are not connected to docking points.

To remove such artifacts, we locally deform some edges near anchoring points to make all anchoring points connected well with their corresponding docking points, respectively. With regard to this, we define an  $M \times M$  window with an unmatched anchoring point as the center, and apply our ICP-based global optimization method to the structures in this window. Note that the deformed structures in this local window should be aligned well not only with the boundary structures, but also with the reference structures near this window. As a result, the new docking points set  $Q$  contains both original docking points in this local window and the intersection points between the edges of the reference structures and the boundary of the window. We add higher weights to the intersection points to enforce that the distortion near the window boundary is small. In general, we set  $M = 21$ , and can get good results in our tests. As illustrated in the green window in Fig. 3d, the reference structures are better aligned with the boundary structures after local refinement.

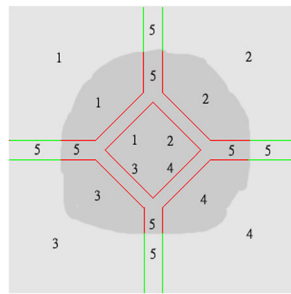
## 5 Image completion

After the structures are restored in the hole, the hole is filled under their guidance. As suggested in [30], we may synthesize images along restored structure information first, and then complete other subregions [15]. Though such a completion is helpful to preserve consistent appearances in the image hole, it is generally expensive, especially when the predicted structures are complex or the hole is large. Considering this, when the reference image has similar appearance with the input image, we adopt Poisson cloning [25] to fast fill the hole, which directly pastes the image patch whose structures can be aligned with the boundary structures, to blend with the surroundings of the hole seamlessly. In the following two subsections, we introduce how to implement these two methods, respectively, in our system.

### 5.1 Completion via statistics of patch offsets

Using the contents in the image itself to complete the hole, it is helpful to preserve consistent appearances. Here, we adopt the fast and effective algorithm [15] which utilizes dominant patch offsets for image completion. As discussed in [30], the restored structures in the hole separate the hole

**Fig. 5** Illustration of the subregions that are separated by restored structures, where the image hole is in gray. Each labeled smaller hole could be filled by the pixels from the known regions labeled in the same number



into smaller holes, and every smaller hole could be filled in similar appearance as their related surroundings of the hole. Thus, the possible pixels for filling the smaller holes could be confined in a smaller region in the image, as illustrated in Fig. 5. In this way, the algorithm can be further accelerated.

## 5.2 Completion via Poisson cloning

Poisson image cloning [25] has been demonstrated an effective approach for seamless image composition.

It is fast and used in [13, 14] for image completion. Unlike them to directly cut a region of the reference image to paste onto the target image, we choose to use the image region not only having similar appearances but also having structures able to align well with the boundary structures, because structure differences between the input and reference images would lead to evident artifacts with Poisson cloning. Here the appearance distance is computed by the  $\chi^2$  distance of color histograms.

To get image regions with plausible structures and similar appearance, we first use the techniques described in Sect. 3 to get many images whose structures could be aligned well with the boundary structures (i.e., top 1000 results), and then rank up these images by SIFT descriptors as suggested in [26], because SIFT descriptors are very effective to represent the image contents, by which the appearance difference is well measured. Afterwards, the top ranked images are used as candidates to fill the image hole.

Figure 6 shows some completion results with Poisson cloning. When the reference image patch has similar appearance with the target image, the completion result will be more plausible, e.g., the result in b seems better than the result in c.

## 6 Results and discussion

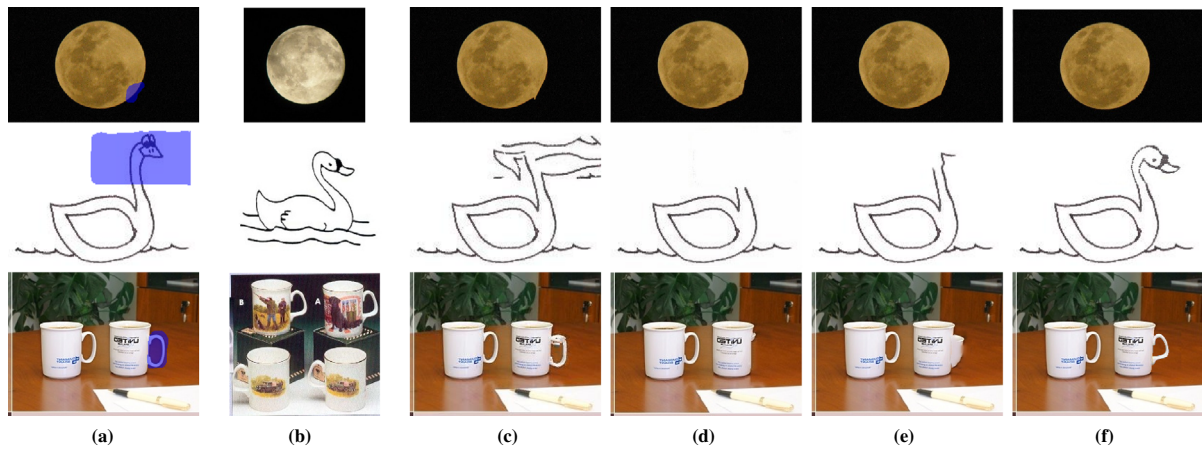
We implemented our method, and made comparison with some well-known methods, including the method for scene completion with image retrieval [14], the Shift-map method for implicitly propagating structures [27], and three methods for explicitly treating structures in image completion, the exemplar-based method by Criminisi et al. [8], the method by tele-registration [18] and the tool of Content-Aware Fill in Adobe Photoshop CS6. Here the tool of Content-Aware Fill is developed based on many techniques [2, 31, 32], especially the PatchMatch technique [2] for treating structure information in image completion. Our tests were performed on a personnel computer installed with an Intel core i7 3.4 GHz CPU and 4G RAM. In the following subsections, we will discuss our approach on its effectiveness, its efficiency and its potential to treat large holes and to enrich the variation of structures in image completion, which is very helpful for creativity design.

### 6.1 Effectiveness

We made tests on restoring simple structures and complex structures, as illustrated in Figs. 7 and 8, respectively. From the results, it is clear that plausible structure restoration is very important to quality image completion, and our approach can achieve this. As for the compared methods, they cannot produce plausible structures in filling the hole, and so making image completion unsuccessful. In particular, the methods for treating structures explicitly in image completion, the method by Criminisi et al. [8] and the tool of Content-Aware Fill, both failed in generating plausible results. Those methods completed images only using struc-

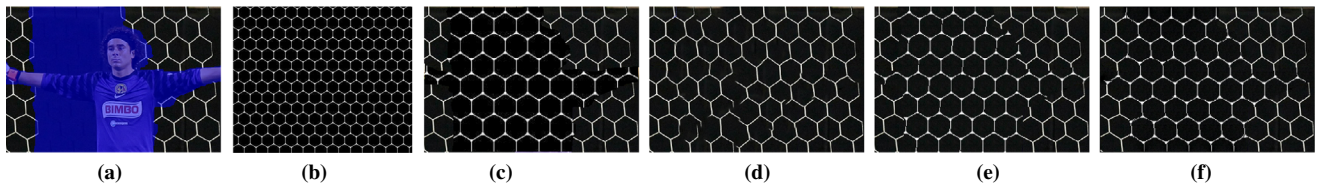


**Fig. 6** Image completions with different reference images. **a** is the input target image with a hole in blue, **b** and **c** are two completions using the first and fifth reference image in **d** (from left to right, and from top to down), respectively. **a** Input image, **b** completion I, **c** completion II and **d** reference images



**Fig. 7** Comparison on restoring simple structures between the method of Criminisi et al. [8], Shift-map [27], Content-Aware Fill and our method, which shows that plausible structure restoration is very important for high-quality image completion. Here, no matter whether the structures are restored implicitly by the Shift-map method or explicitly

by the methods of Criminisi et al. and Content-Aware Fill, they did not generate plausible structures in image completion. **a** Input image, **b** reference image, **c** Criminisi et al.'s [8], **d** Shift-map [27], **e** Content-Aware Fill and **f** our result



**Fig. 8** Both reference structures and structure restoration are essential for plausible image completions. Without structure restoration, scene completion method [14] cannot match structures well (c). Content-Aware Fill may be hard to predict plausible structures in the image

hole without reference images (d). Method [18] cannot align structures well (e). Our method can produce satisfactory results. **a** Input image, **b** reference image, **c** scene completion, **d** Content-Aware Fill, **e** result of [18] and **f** our result

tures in source image, when the source images cannot provide enough information to keep suitable structure propagation, those methods may fail. This is also one of the reasons why it is a hot topic recently to use manually drawn strokes or sketches to specify structures explicitly in the hole for image completion.

When images have complex structures, we argue both reference structures and structure propagation are essential for plausible image completions, as shown in Fig. 8. Without reference structures, Content-Aware Fill and method [18] cannot align the structures well. This is because it is generally difficult to predict plausible structures to blend well with the complex surrounding structures in the hole. Though Scene Completion [14] used reference images to complete scene images, it cannot guarantee that structures are aligned well, which may cause artifacts, as shown in Fig. 8c. Our method use reference structures for more useful information and apply structure propagation for effective structure alignment, thus get better results.

Generally speaking, the methods with manually drawn strokes or sketches to specify structures in the hole can well restore plausible structures. But it is time consuming on draw-

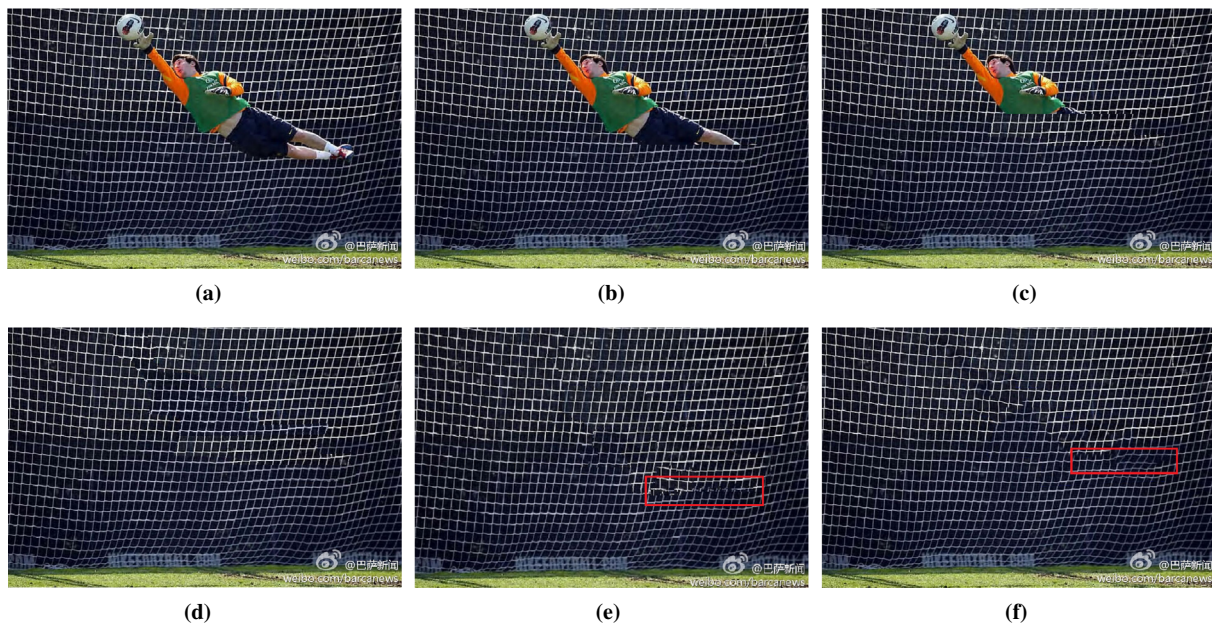
ing, not helpful for efficient image completion, which will be discussed in the following subsection. As for our method, in some sense, we explicitly copy some structures to fill the hole, which takes the same effect of the specified structures by strokes or sketches. Thus it is guaranteed that we can restore structures plausibly in image completion.

### 6.2 Large holes

It is always a challenge to fill a large hole, as it is generally difficult to get large structures. For this, we take the following strategy for filling a large hole, partitioning the large hole into smaller ones, and filling the smaller holes gradually. In general, we initially divide a hole into two smaller ones, with one about one-third of the entire hole, and the other about two-thirds, and they share a small band region for blending. When a smaller hole cannot be completed effectively, it will be partitioned and treated iteratively in the same manner until the obtained small holes can be well filled.

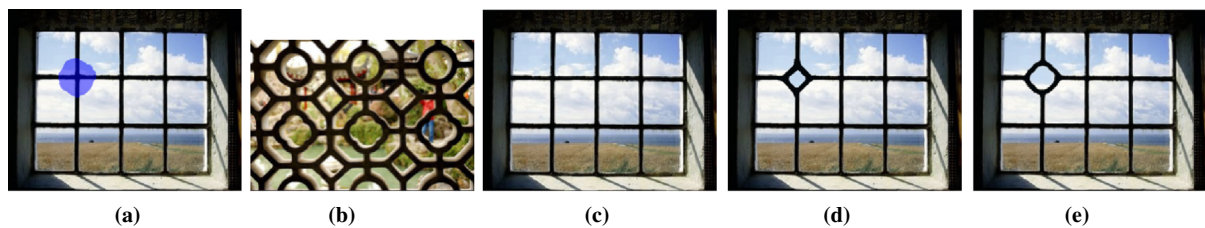
As illustrated in Fig. 9, when we want to remove the footballer to complete the net a, we can complete the large hole by treating its divided smaller holes gradually from b through





**Fig. 9** Image completion with large holes filled. Our method filled smaller holes gradually from **b** through **c** to **d** in the final to complete the large hole, where we used input image itself as the reference image. Our result **d** can better align reference structures than Shift-map [27] (**e**)

and Content-Aware Fill (**f**), as shown in the *red boxes*. **a** Input image, **b** fill the bottom part, **c** fill the middle part, **d** our final result, **e** Shift-map [27] and **f** Content-Aware Fill



**Fig. 10** Our system can generate many completions easily. **a** is the input image with a hole in *blue*, **b** is a reference image for providing plausible structures, and **c**, **d** and **e** are completion results with reference

structures from the image **a** and **b**. **a** Input image, **b** reference image, **c** completion I, **d** completion II and **e** completion III

c to d finally. Compared with the completions of other methods such as Shift-map and Content-Aware Fill, our result is much better without evident artifacts, while the others have unconnected edges or mismatched edges in the restored structures.

This is an advantage of our approach over the methods with manual strokes or sketches to specify structures, because it is very trouble or time consuming to specify plausible structures by manually drawing strokes or sketches in the large hole.

### 6.3 Structure variation

Clearly, with our system, more plausible structures can be provided for the user to choose for image completion. This is benefited from the tremendous image resources on the internet. Thus, the user has more chances to generate his

desired results, which is very helpful for creativity design. As illustrated in Fig. 10, the image hole in the image **a** can be completed with the structures from the image itself, or from a reference image **b** to get the completed images **c**, **d** and **e**, respectively. Clearly, these adopted structures are all acceptable for human beings, where the method via statistics of patch offsets [15] was used to fill the other regions of the hole. We also provided three completions of the mug example in Figs. 6 and 7.

### 6.4 Efficiency

With our system to fill a hole, we can save much time on measuring the consistency between nearby patches, because we take the reference structures as a whole to fill the hole. Thus, we can run faster than many patch-based methods. As illustrated in Table 1, where we list the statistics on time cost



**Table 1** Time cost of our system and other methods

	Moon (Fig. 7)	Swan (Fig. 7)	Mug (Fig. 7)	Goalkeeper (Fig. 8)	Footballer (Fig. 9)	Window (Fig. 10)
Image size (pixels)	400 × 240	197 × 149	265 × 199	293 × 198	955 × 630	422 × 325
Percentage of the hole in the image (%)	1.4	22.2	5.9	62.8	13.8	2.8
Time cost in the steps of our system						
Structure retrieval	0.056	0.051	0.055	0.075	0.113	0.084
Localization (LTVCI0)	0.712	0.392	0.443	2.572	54.416	0.543
Initial transformation (HPRY09)	0.064	0.064	0.062	0.081	0.625	0.654
Structure propagation	0.013	0.045	0.021	0.752	6.251	0.324
Completion	0.121	0.424	0.341	0.521	0.642	0.315
Total time of our system	0.966	0.976	0.922	4.001	62.047	1.92
Completion by Criminisi et al.'s	1.618	3.642	0.998	34.706	1219.258	8.639
Completion by Shift-map	6	18	10	14	20	6
Completion by Content-Aware Fill	<1				1.7	<1

Note that the time cost of Content-Aware Fill is estimated by the use of Adobe Photoshop CS6 (seconds)

of all methods, our system is much faster than the method by Criminisi et al. [8] and the Shift-map method [27] except that the Shift-map method treating the large hole in Fig. 9, where our system divided the large hole into smaller holes to fill progressively, causing our system to run many times.

Here, we did not compare our approach and the methods with manually drawn strokes [30] or sketches [13] to specify structures. With a simple thought, it could be known that our approach is more efficient in general. This is because drawing strokes or sketches to specify structures always need at least several seconds, while we can use at most several seconds for a simple filling according to the statistics in Table 1.

From the statistics in Table 1, ours is a little slower than the tool of Content-Aware Fill. The tool is developed in a system, taking full advantages of the hardware and software. As for our system, it is just a prototype. To promote the working efficiency of our system, many works will be done. For example, the localization step takes the majority of our time cost in our current implementation, and this can be improved by running this step in parallel.

### 6.5 Limitation

Our system depends on the extracted edges for structure retrieval, which may fail in some cases when the edge information is from shadow or something else instead of structure information. With regard to this, we need further study how to effectively obtain structure information.

### 7 Conclusion

Existing image completion methods often produce unsuitable structure connection between neighboring regions, mostly because there are not enough candidates provided. For this, we design a novel method to conveniently get suitable structures for hole filling from the internet resources. To consistently connect the retrieved structures with the structures around the hole, another aspect for reducing artifacts in structure restoration, we develop a new energy function for optimizing the connection, superior to existing techniques. Thus, we can fill holes well and efficiently, even with complex structures.

As a benefit of our approach, we can easily and fast fill large holes and have the potential of providing many plausible structures for creativity design. This is superior to the methods using manually drawn strokes or sketches to specify the structures in the hole, as they are generally time consuming and not easy to predict many plausible structures, especially in filling large holes.

**Acknowledgments** This work is partially supported by National Natural Science Foundation of China (61379087), the Knowledge Inno-

vation Program of the Chinese Academy of Sciences and the European Unions Seventh Framework Programme (FP7/2007-2013) under grant agreement (612627).

## References

1. Bar, M.: Visual objects in context. *Nat. Rev. Neurosci.* **5**(8), 617–629 (2004)
2. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: Patchmatch: a randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.* **28**(3), 24:1–24:11 (2009)
3. Bertalmio, M., Vese, L.A., Sapiro, G., Osher, S.: Simultaneous structure and texture image inpainting. *IEEE Trans. Image Process.* **12**(8), 882–889 (2003)
4. Besl, P.J., McKay, N.D.: A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(2), 239–256 (1992)
5. Chen, Y., Medioni, G.: Object modelling by registration of multiple range images. *Image Vis. Comput.* **10**(3), 145–155 (1992)
6. Cheng, M.M., Zhang, F.L., Mitra, N.J., Huang, X., Hu, S.M.: Repfinder: finding approximately repeated scene elements for image editing. *ACM Trans. Graph.* **29**, 83:1–83:8 (2010)
7. Chia, A.Y.S., Zhuo, S., Gupta, R.K., Tai, Y.W., Cho, S.Y., Tan, P., Lin, S.: Semantic colorization with internet images. *ACM Trans. Graph.* **30**(6), 156:1–156:8 (2011)
8. Criminisi, A., Pérez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. Image Proc.* **13**(9), 1200–1212 (2004)
9. Darabi, S., Shechtman, E., Barnes, C., Goldman, D.B., Sen, P.: Image melding: combining inconsistent images using patch-based synthesis. *ACM Trans. Graph.* **31**(4), 82:1–82:10 (2012)
10. Eitz, M., Hays, J., Alexa, M.: How do humans sketch objects? *ACM Trans. Graph.* **31**(4), 44:1–44:10 (2012)
11. Eitz, M., Hildebrand, K., Boubekur, T., Alexa, M.: Sketch-based image retrieval: benchmark and bag-of-features descriptors. *IEEE Trans. Vis. Comput. Graph.* **17**(11), 1624–1636 (2011)
12. Farbman, Z., Fattal, R., Lischinski, D., Szeliski, R.: Edge-preserving decompositions for multi-scale tone and detail manipulation. *ACM Trans. Graph.* **27**(3), 67:1–67:10 (2008)
13. Goldberg, C., Chen, T., Zhang, F.L., Shamir, A., Hu, S.M.: Data-driven object manipulation in images. *Comput. Graph. Forum* **31**(2), 265–274 (2012)
14. Hays, J., Efros, A.A.: Scene completion using millions of photographs. *ACM Trans. Graph.* **26**(3) (2007)
15. He, K., Sun, J.: Statistics of patch offsets for image completion. In: *ECCV'12*, pp. 16–29 (2012)
16. Ho, J., Peter, A.M., Rangarajan, A., Yang, M.H.: An algebraic approach to affine registration of point sets. In: *ICCV*, pp. 1335–1340 (2009)
17. Hu, S.M., Chen, T., Xu, K., Cheng, M.M., Martin, R.: Internet visual media processing: a survey with graphics and vision applications. *Vis. Comput.* **29**(5), 393–405 (2013)
18. Huang, H., Yin, K., Gong, M., Lischinski, D., Cohen-Or, D., Ascher, U., Chen, B.: “Mind the gap”: tele-registration for structure-driven image completion. *ACM Trans. Graph.* **32**(6), 174:1–174:10 (2013)
19. Jia, J., Tang, C.K.: Image repairing: robust image synthesis by adaptive and tensor voting. In: *CVPR*, pp. 643–650 (2003)
20. Komodakis, N.: Image completion using global optimization. In: *CVPR*, pp. 442–452 (2006)
21. Liu, M.Y., Tuzel, O., Veeraraghavan, A., Chellappa, R.: Fast directional chamfer matching. In: *CVPR*, pp. 1696–1703 (2010)
22. Mansfield, A., Prasad, M., Rother, C., Sharp, T., Kohli, P., Gool, L.J.V.: Transforming image completion. In: *BMVC*, pp. 1–11 (2011)
23. Mu, T.J., Wang, J.H., Du, S.P., Hu, S.M.: Stereoscopic image completion and depth recovery. *Vis. Comput.* **30**(6–8), 833–843 (2014)
24. Nill, N.B., Bouzas, B.H.: Objective image quality measure derived from digital image power spectra. *Opt. Eng.* **31** (1992)
25. Pérez, P., Gangnet, M., Blake, A.: Poisson image editing. In: *SIGGRAPH*, pp. 313–318 (2003)
26. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: *CVPR* (2007)
27. Pritch, Y., Kav-Venaki, E., Peleg, S.: Shift-map image editing. In: *ICCV*, pp. 151–158 (2009)
28. Simakov, D., Caspi, Y., Shechtman, E., Irani, M.: Summarizing visual data using bidirectional similarity. In: *CVPR*, pp. 1–8 (2008)
29. Sivic, J., Zisserman, A.: Video google: a text retrieval approach to object matching in videos. In: *ICCV*, pp. 1470–1477 (2003)
30. Sun, J., Yuan, L., Jia, J., Shum, H.Y.: Image completion with structure propagation. In: *SIGGRAPH*, pp. 861–868 (2005)
31. Wexler, Y., Shechtman, E., Irani, M.: Space-time video completion. In: *CVPR*, pp. 120–127 (2004)
32. Wexler, Y., Shechtman, E., Irani, M.: Space-time completion of video. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(3), 463–476 (2007)



**Miao Hua** received his B.S. degree from University of Science and Technology of China in 2009. He is currently a Ph.D. candidate at the State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences. His research interests include image and video processing and computer graphics.



**Wencheng Wang** is a professor of the State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, where he leads a research group on Computer Graphics and Image Processing. He received his Ph.D. degree in software from Institute of Software, Chinese Academy of Sciences in 1998. His research interests include visualization, image editing and virtual reality. He is a member of the IEEE and the ACM.