

RESEARCH

Open Access



# Image completion via transformation and structural constraints

Qiaochuan Chen<sup>1</sup>, Guangyao Li<sup>1\*</sup>, Qingguo Xiao<sup>1</sup>, Li Xie<sup>1</sup> and Mang Xiao<sup>2</sup>

\* Correspondence: [lgy423@126.com](mailto:lgy423@126.com)

<sup>1</sup>College of Electronics and Information Engineering, Tongji University, Shanghai, China  
Full list of author information is available at the end of the article

## Abstract

Image completion is an approach to fill a damaged region (hole) in an image. In this study, we adopt a novel method which can repair a target region with structural constraints in an architectural scene. An objective function that consists of three terms is proposed to solve the image completion problem. In color term, we compute a parameterized transformation model using detected plane parameters and measure the distance between the target patch and transformed source patch. This model helps to extend the patch search space and find an optimal solution. To improve the patch matching accuracy, we add a guide term that includes structure term and consistency term. The structure term encourages sampling patches along the structural direction, and the consistency term is used to maintain the texture consistency. Considering the color deviation between patches, we add a gradient term into a framework that can solve more challenging problems. Compared with previous methods, the proposed method has good performance in preserving global structure and reasonably estimating perspective distortions. Moreover, we obtain acceptable results in natural scenes. The experimental results illustrate that this novel method is a potential tool for image completion.

**Keywords:** Image completion, Image inpainting, Structural constraints, Transformation, Objective function

## 1 Introduction

Image completion methods aim to repair the defects of digital images with plausibly synthesized content to make images look more natural. This task is applied to many image editing applications ranging from object removal to movie clip and image understanding [1–3]. In general, there are two main types of image completion methods: diffusion-based methods and exemplar-based methods.

A diffusion-based method completes the target region using partial differential equations which propagate image information from surrounding areas into an unknown region. Bertalmio et al. [4] first proposed a method in which the information was propagated through the edge of a contour line in the occlusion area. Furthermore, these methods have two types: Euler's model [5] and total variation model [6]. They perform well in the images with thin cracks and scratches; however, they are not suitable for large damaged regions.

Exemplar-based methods sample the pixels from a known region of the image and copy them to a damaged region. Efros and Leung [7] proposed a non-parametric method for texture synthesis. The texture synthesis process grows a new image outward from an initial seed, one pixel at a time. The method in Ref. [8] processed an image in a greedy way to research the best matching patches. Due to the greedy strategy, this method resulted in an inconsecutive texture. Sun et al. [9] developed a method that first allowed users to draw lines in a target region. Then, the target region was completed along the lines. Meanwhile, some approaches [10, 11] improve Criminisi's effect and pose the completion task as a global optimization problem with a well-defined objective function and propose an algorithm to optimize it. However, the cost of the objective function is usually exorbitant. According to this case, a fast PatchMatch method [12] solves this problem to a considerable extent by propagating the neighborhood information using neighbor patches. Furthermore, this method has been adopted by Adobe Photoshop. As for simple patch translation, it is difficult to find the most suitable patch without extending the search space. In fact, many methods [13–17] have addressed this issue via geometric transformation as well as photometric transformation. Xiao et al. [18] filled in the target region using a sample image. This method adopted an image with a similar texture and structure to enrich the search space. Le Meur et al. [19] used a coarse version of the input image to generate multiple inpainted images with different parameter settings and then recovered the full resolution of the final result. Using a Markov Random Field (MRF) model to build the energy function, many methods [20, 21] optimize the energy function for its efficiency in realizing global image consistency. He and Sun [22] calculate the statistical offset to obtain regular structure information. This method demonstrates an excellent result in images with a large amount of duplicated information; however, it is still a problem when images have perspective shape deformation. The methods in Refs. [23, 24] using a convolutional neural network (CNN) to generate the contents according to its surroundings. This method provided a great solution for filling in a large region and keeping the image semantically correct. However, it could not handle images with perspective distortion.

In this study, we propose a novel method for inpainting damaged regions in structural scenes. We also extend the method to different natural scenes. It has been discovered that most of the texture in various scenes have structured features (regular or linear). Therefore, we detect these parameters and make use of them to find a transformational relation between source patch and target patch. Moreover, we propose an objective function with two constraints to guide the texture synthesis. Different from previous methods, these two constraints can provide effective guidance when searching for the best matching patches. Finally, we apply a gradient term that is conducive to a gradual adjustment of the colors of our objective function to maintain the texture details.

The three main contributions of the proposed method can be described as follows. First, we adopted a parameterized transformation model to guide the image completion process. Second, we proposed an objective function with two constraints which together guide texture synthesis. Third, we combine the effect of these constraints and gradients into a framework that solves more challenging problems.

### 2 Method overview

Given an input image  $I$  with a damaged region (hole), we aim to fill in the damaged region using pixels from the known region. In practice, it is a challenge to fill in a damaged region and obtain satisfying results, especially in architecture scenes. In many real scenes, the shape can change dramatically because of perspective distortions. For each target patch  $P$  in the damaged region, we calculate a transformation matrix  $T_i$  that correlate target patch to the best matching patch  $Q$ . To estimate the parameters of  $T_i$ , we adopt detected plane parameters [25] to generate the transformation model, instead of searching for the best matching patches by simply translation (details given in Section 3.1). When searching for the similar source patches, unconstrained process usually causes poor results. We constrain the patch sampling locations using texture direction and texture consistency (Section 3.3). Furthermore, we add gradient into our framework to obtain a smooth transition of color.

To obtain plausible texture in the hole region, the problem is translated into an optimization scheme. We define an objective function consists of color term, guide term, and gradient term. The color term explains how the source patches should be transformed. The guide term provides constraint, i.e., how the searching process should be limited. The gradient term gives an adjustment that leads to a smooth transition of color. Combining these three terms, we show that the proposed method can effectively improve the completion results in visual consistency. The flowchart of the proposed is shown in Fig. 1.

### 3 Objective function

To achieve a high-quality result, we develop an objective function for image completion. The objective function is a measured distance function that includes three terms. Here, we develop a transformation parametrized by  $\theta_i$  for each patch  $P$ .

We denote the improved energy minimization function as follows:

$$E = \sum_{i \in \Omega} E_{\text{color}}(s_i, t_i, \theta_i) + E_{\text{guide}}(s_i, t_i) + E_{\text{gradient}}(s_i, t_i, \theta_i), \tag{1}$$

where  $t_i = (t_i^x, t_i^y)^T$  is the center position of a target patch in  $\bar{\Omega}$  and  $s_i = (s_i^x, s_i^y)^T$  is the

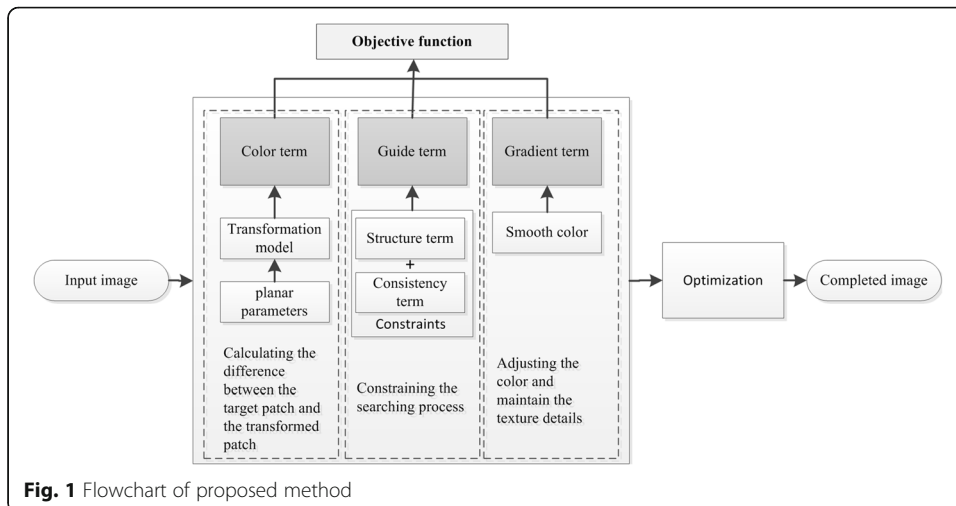


Fig. 1 Flowchart of proposed method

center position of the corresponding source patch in  $\Omega$ . Here,  $\Omega$  and  $\bar{\Omega}$  are the labels of known pixels and unknown pixels, respectively. We define  $\theta_i$  as a set of parameters for generating a transformation matrix  $T_i$ . The three terms  $E_{color}$ ,  $E_{gradient}$  and  $E_{guide}$  are the color term, gradient term and guide term, which together form the function. These terms will be explained in detail in the following sections.

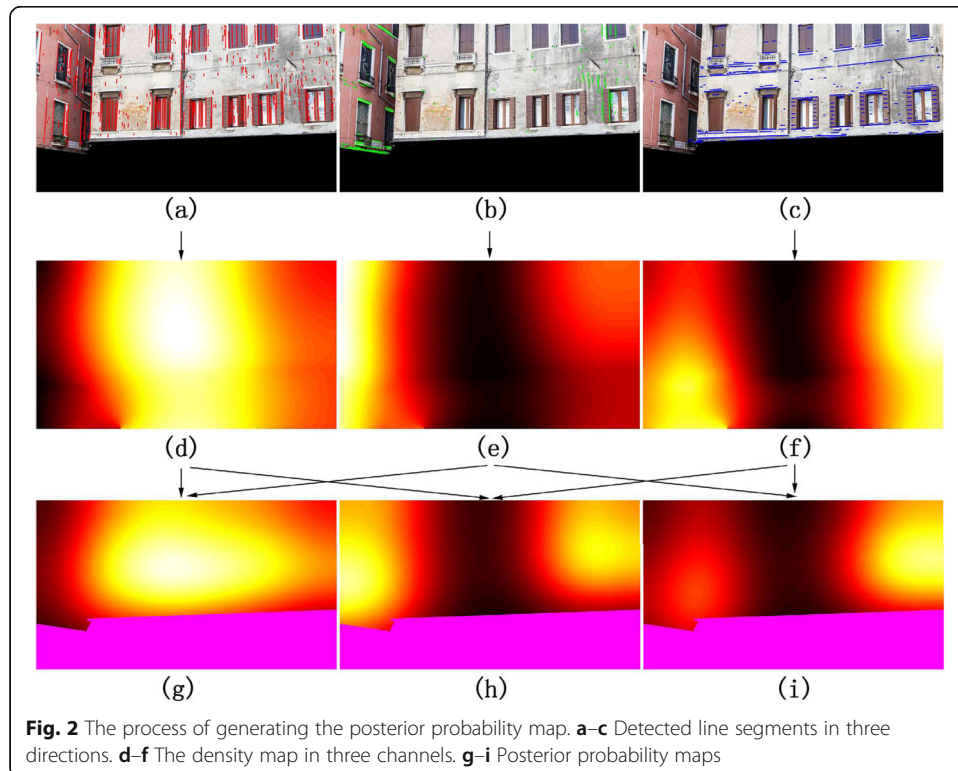
### 3.1 Color term

The color matching term is similar to Ref. [8]:

$$E_{color}(t_i, \theta_i) = \|P(t_i^x, t_i^y) - Q(s_i^x, s_i^y, \theta_i)\|_2^2, \tag{2}$$

where  $P(t_i^x, t_i^y)$  is the target patch centered at  $t_i$ , and  $Q(t_i^x, t_i^y, \theta_i)$  denote the matched source patch using the transformation matrix  $T_i$  with the parameter  $\theta_i$ . Here, the color term represents the distance between the target patch and the transformed patch. We use sum of squared distance in the RGB space to calculate the distance. In Refs. [13–15], many geometric transformations were applied, e.g., rotation, scale, and flip. On the contrary, we use a homograph matrix to transform the patches into an affine correction space.

We now illuminate how we generate the transformation matrix  $T_i$  based on the parameter  $\theta_i$ . In many real scenes, the shape can change dramatically because of a perspective distortion. It is difficult to fill in a damaged region if only simple patch transformation is taken into consideration. Xiao et al. [25] solved this problem by detecting planes and making use of them to generate a projective transformation matrix. In Fig. 2, we show the plane detection and posterior probability map.



**Fig. 2** The process of generating the posterior probability map. **a–c** Detected line segments in three directions. **d–f** The density map in three channels. **g–i** Posterior probability maps

In our paper, we use the detected planar parameters to parametrize  $T_i$  by  $\theta_i = (f_i, k_i)$ , where  $k_i$  is the index of plane and  $f_i = (f_i^x, f_i^y, f_i^s, f_i^\theta, f_i^\alpha, f_i^\beta)$  is the six-dimensional affine parameter. We define a transformation matrix as follow:

$$T_i(l_\infty^k, t_i, f_i, \theta_i) = H_p(l_\infty^k, t_i, f_i, k_i)H_r(f_i^\theta)H_s(f_i^s)H_c(f_i^\alpha, f_i^\beta)H_t(f_i^x, f_i^y), \quad (3)$$

where  $H_p$  indicates the projective transformation between source patch and target patch. The matrix  $H_p$  has the form:

$$H_p = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1^k & l_2^k & l_3^k \end{bmatrix} \quad (4)$$

here,  $l_\infty^k = (l_1^k, l_2^k, l_3^k)$  is vanishing line which has two degrees. The matrix  $H_r$ ,

$$H_r = \begin{bmatrix} M(f_i^\theta) & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (5)$$

indicates a rotation transformation by a  $2 \times 2$  rotation transformation  $M(f_i^\theta)$ . We define the matrix  $H_s$  as follow:

$$H_s = \begin{bmatrix} N(f_i^s) \\ 0 \end{bmatrix} = \begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (6)$$

where  $H_s$  indicates the scale transformation by a  $2 \times 2$  scaling transformation  $N(f_i^s)$ . The matrix  $H_c$

$$H_c = \begin{bmatrix} 1 & f_i^\alpha & 0 \\ f_i^\beta & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (7)$$

indicates the shear transformation. The matrix  $H_t$

$$H_t = \begin{bmatrix} 1 & 0 & f_i^x \\ 0 & 1 & f_i^y \\ 0 & 0 & 1 \end{bmatrix}, \quad (8)$$

indicates the translation transformation by translation parameters  $f_i^x$  and  $f_i^y$ . The transformation model is similar to the decomposition of projective transformation matrix [26]. This formula effectively shows the transformation relation between source patch and target patch.

### 3.2 Guide term

Owing to the difficulty of acquiring excellent inpainting results just using color and gradient, we apply a guide term to constrain the patch search. Our guide term includes two constraints:

$$E_{guide}(s_i, t_i) = \lambda E_{structure}(s_i, t_i) + E_{consistency}(s_i, t_i), \quad (9)$$

where  $\lambda$  is the weight of the structure term. These two constraints can together guide the completion process.

### 3.2.1 Structure term

Many approaches [12, 27, 28] have demonstrated that limiting the search space by labeling the texture region could improve the completion result. Hence, we adopt a method using Gray-level co-occurrence matrix (GLCM) to detect the dominant texture direction and then automatically generate a structure guidance map that serves as a position constraint. The detail about this method can refer to Ref. [28]. In this study, we improve this method by further analyzing the optimal direction angle.

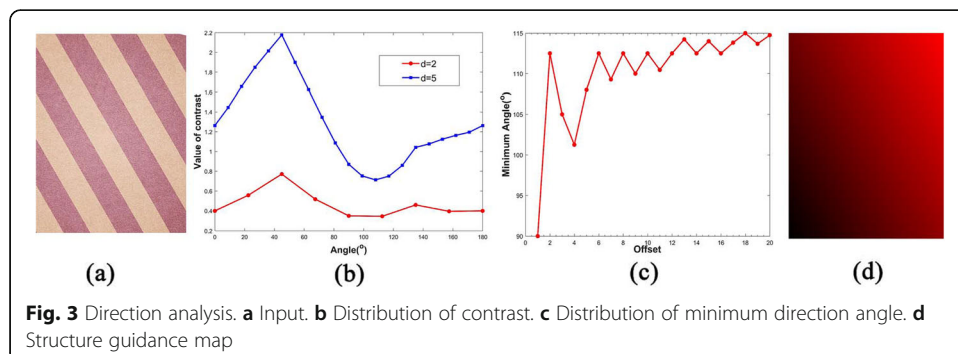
Based on Ref. [28], the greater the GLCM contrast, the smaller the similarity between two pixels. We also obtain the relation between offset value ( $d$ ) and the number of direction angle: the greater the offset value, the more is the directional angles. Zarif et al. [28] analyzed the texture direction using eight direction angles ( $d = 2$ ). In this study, we compute the minimum of contrast to detect the current direction angle (also called minimum direction angle), as shown in Fig. 3b. We detect more directions to determine the optimal direction. Note that big value of offset may reduce the sensitivity to the texture direction. Thus, we set the maximum of offset  $d_{\max} = 20$ . The distribution of minimum direction angle is illustrated in Fig. 3c. We adopt the average value of all the minimum direction angle to determine the optimal texture direction.

Given an original image, the content along a direction usually has a similar structure and texture. To develop this property, we use the detected optimal direction to represent the content changes. Rather than limiting the search space using a non-gradient color, a gradient color is adopted, as shown in Fig. 3d. Here, the structure guidance map is regarded as a soft constraint in the completion process. In the structure guidance map, the location of the same color usually has the same texture. The structure guidance map encourages searching similar patches along the same direction.

The structure term  $E_{\text{structure}}$  makes use of the guidance map to constrain the position where the source patches are drawn from ( $G_{\text{pos}}$ ). The structure term is defined as follows:

$$E_{\text{structure}}(s_i, t_i) = L(|G_{\text{pos}}(s_i^x, s_i^y) - G_{\text{pos}}(t_i^x, t_i^y)|), \quad (10)$$

where  $L(\cdot)$  is the  $\epsilon$ -insensitive loss function  $L(x) = \max(0, x - \epsilon)$ . We denote  $G_{\text{pos}}$  as the position information of the source patch and target patch in structure guidance map.  $G_{\text{pos}}$  indicates the pixel values at the center of sampling patches. According to GLCM, locations that have the same color in structure guidance map usually have similar textures. This means that locations with similar pixel values ( $G_{\text{pos}}$ ) in structure guidance map often have similar textures. Thus, the source patch and target patch have different



values in  $G_{\text{pos}}$  should be penalized by this term. Sampling along the texture direction is encouraged to minimize the energy function.

### 3.2.2 Consistency term

Inspired by Ref. [12, 20], we add a consistency term into the completion process to sample patches in adjacent regions. Given a target patch  $t_i$ , if we can find a matching patch  $s_i$ , their neighboring patches  $t_i^n$  and  $s_i^n$  are very likely to be the most similar patches. We assume that every patch has neighbors in four directions. We define  $t_i^{n_1}$ ,  $t_i^{n_2}$ ,  $t_i^{n_3}$ , and  $t_i^{n_4}$  as the neighboring patches of  $t_i$  and  $s_i^{n_1}$ ,  $s_i^{n_2}$ ,  $s_i^{n_3}$ , and  $s_i^{n_4}$  as the neighboring patches of  $s_i$ , respectively. If the difference between neighboring patches exceeds a threshold, we add a consistency constraint to encourage sampling patches from neighboring areas. The consistency term has the form:

$$E_{\text{consistency}}(s_i, t_i) = \sum_{j=1,2,3,4} [ |C_{s_i}^{n_j}(s_i^x, s_i^y) - C_{t_i}^{n_j}(t_i^x, t_i^y)| > \varepsilon ], \quad (11)$$

where  $C_{s_i}^{n_j}$  and  $C_{t_i}^{n_j}$  represent the current position of the neighboring patches.  $C_{s_i}^{n_j}$  indicates the distance between a target patch and its neighboring patch. Similarly,  $C_{t_i}^{n_j}$  indicates the distance between a source patch (corresponding to the target patch) and its neighboring patch. If the argument is false, the indicator function  $[\cdot]$  is 0; otherwise, it is 1.  $C_{s_i}^{n_j}$  and  $C_{t_i}^{n_j}$  have large difference value should be penalized. If the difference value  $|C_{s_i}^{n_j}(s_i^x, s_i^y) - C_{t_i}^{n_j}(t_i^x, t_i^y)| > \varepsilon$ , the argument is true and  $[|C_{s_i}^{n_j}(s_i^x, s_i^y) - C_{t_i}^{n_j}(t_i^x, t_i^y)| > \varepsilon] = 1$ . Similarly, if the difference value  $|C_{s_i}^{n_j}(s_i^x, s_i^y) - C_{t_i}^{n_j}(t_i^x, t_i^y)| \leq \varepsilon$ , the argument is false and  $[|C_{s_i}^{n_j}(s_i^x, s_i^y) - C_{t_i}^{n_j}(t_i^x, t_i^y)| > \varepsilon] = 0$ . Here, we set  $\varepsilon = 1$  to encourage sampling near the source patch. It helps to maintain the texture consistency.

### 3.3 Gradient term

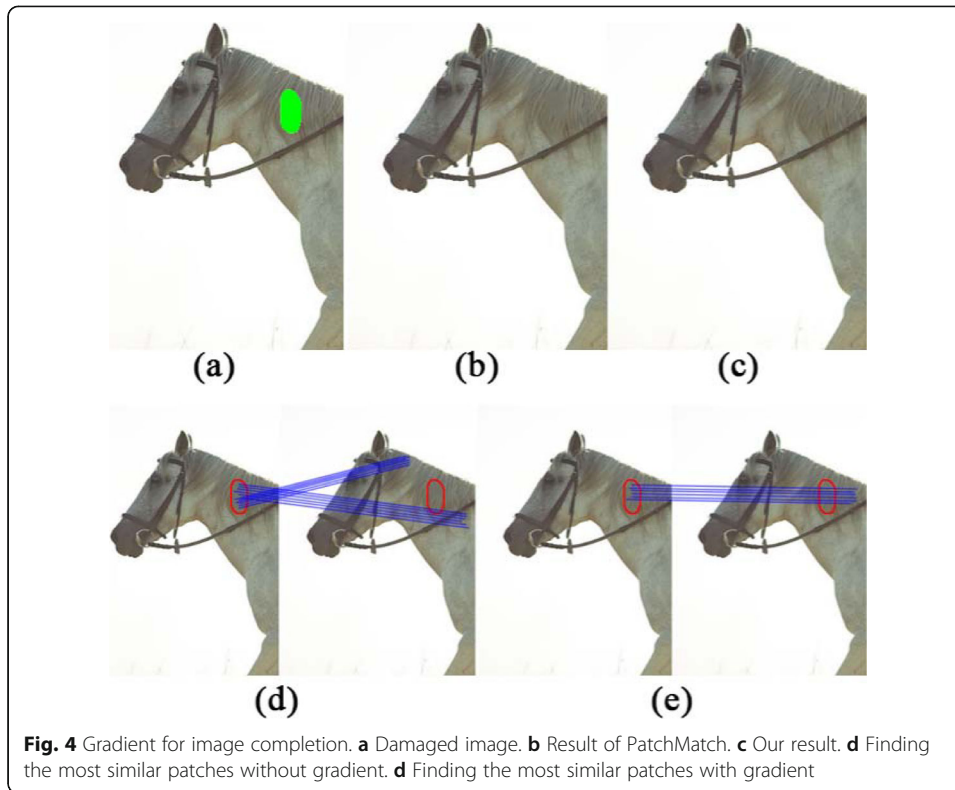
To improve the results of completion, finding correct patches is necessary. Barnes et al. [12] adopted  $L_2$  patch distance to compute the similarity between two patches. However, PatchMatch [12] may discover patches incorrectly when the texture is complicated, as shown in Fig. 4d. The method of Barnes fails to find the correct texture because it does not consider gradient. Adding a gradient term is helpful for gradually adjusting the colors. We define the gradient term as follows:

$$E_{\text{gradient}} = \|\nabla Q(s_i, t_i, \theta_i) - \nabla P(t_i)\|_2^2, \quad (12)$$

where  $\nabla Q(s_i, t_i, \theta_i)$  and  $\nabla P(t_i)$  denote the gradient of the patches centered at  $s_i$  and  $t_i$ , respectively. The gradient term is used to adjust the local color of patch. It can lead to a globally smooth transition of intensity and color [14]—a property that is lacking in patch-based methods. Here, we also use sum of squared distance in the RGB space to calculate the distance. This term can play to our strengths and search for the best similar patch for higher consistency.

## 4 Optimization

Given a large search space, it is intractable to discover a globally optimal completion. Wexler et al. [11] proposed an iterative algorithm which includes two steps named search and voting to optimize an objective function.



**Fig. 4** Gradient for image completion. **a** Damaged image. **b** Result of PatchMatch. **c** Our result. **d** Finding the most similar patches without gradient. **e** Finding the most similar patches with gradient

In the search step, we adopt PatchMatch [12] method to accelerate our algorithm. When searching for the best matching patches, the position of a matching patch is found first. Then, we search for a transformed matching patch. The nearest neighbor patches are searched in the source region for every target patch to minimize the function.

Unlike previous methods, we reject unlikely patch transformation in scale when finding the similar patches, i.e.,  $scale_1 \leq S_{scale}(T_b, s_b, t_i) \leq scale_2$ , where  $S_{scale}(T_i)$  indicates the scale estimation. Large range of scale cannot provide effective constrain when finding source patches. Too small range of scale can lead to narrow patch searching space. We set  $scale_1 = 0.7$  and  $scale_2 = 1.3$  as the acceptable range in our experiments and obtain valid results. The approximated scale can be estimated using the first-order Taylor expansion [29].

In the voting step, the overlapping patches containing  $p$  have correspondence patches in the source region. Wexler et al. [11] adopted a weighted voting program to fill a target region. Similarly, we take the median of all the votes as the pixel to reduce the blur of pixel colors.

When calculating the patch distance, following HaCohen et al. [30], bias and gain are added to obtain the best matching patches. In this study, we set bias to  $[-50, 50]$  and gain to  $[0.5, 1.5]$ . They are used to reject source patches whose gain or bias deviates the range. This can also help to extend the patch searching space and match wide color difference.

## 5 Experimental results and discussion

### 5.1 Implementation details

Our algorithm was implemented with MATLAB and C++. The PatchMatch iteration was [20, 30]. A large hole region required more iterations. The time of the proposed

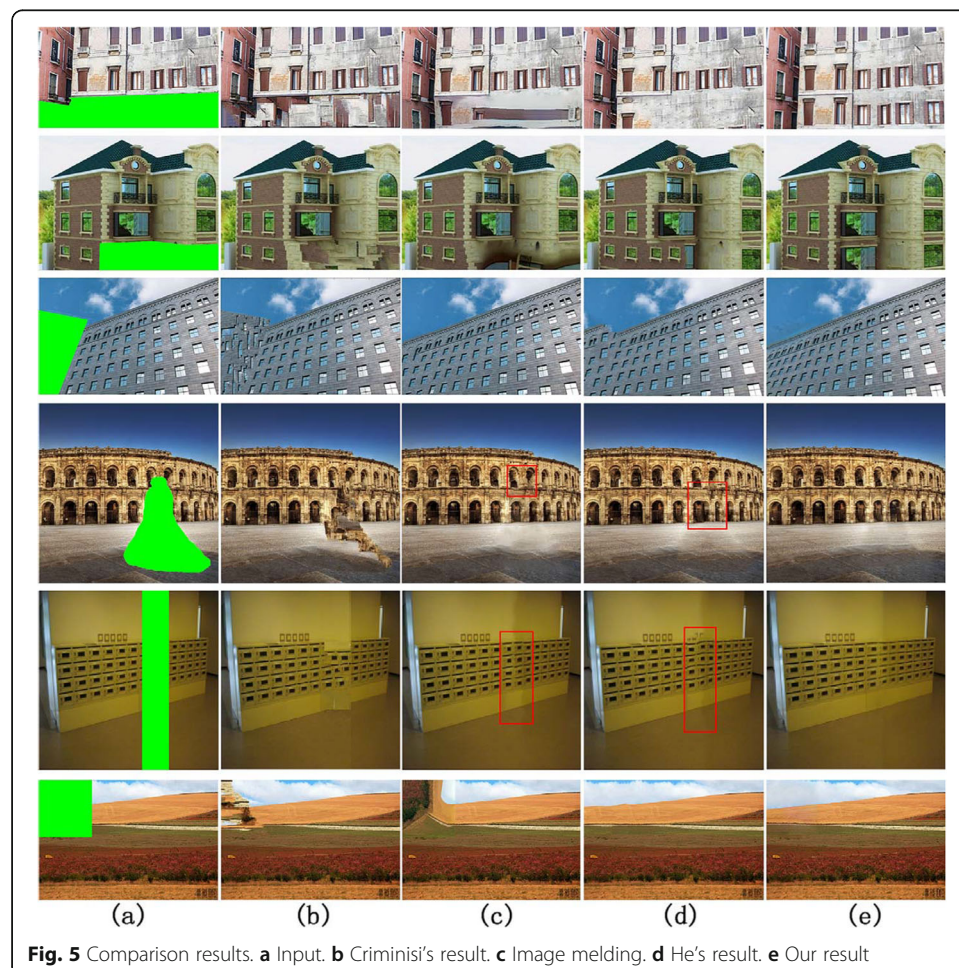


image completion method can be categorized into two cases. The first case is to generate several guidance maps, which requires several seconds. The second case, which determines the running time, depends on the image size, hole region, and the texture complexity. For instance, given a  $400 \times 600$  image with  $120 \times 140$  damaged region, the inpainting process may require 2–3 min.

## 5.2 Comparison results

To demonstrate the results of the proposed method, we compare our method with several existing image completion algorithms, including Criminisi [8], image melding [14] and He and Sun [22]. We run these methods on six test images, as shown in Fig. 5.

In the first two rows, the buildings contain more than one plane. We can see that the proposed method can deal well with structural scenes. The other methods could not maintain structural consistency if only using patch translation. In the third row, we show buildings with projective distortions. Criminisi's method obviously propagated error information into a damaged region because of the flaw of priority in special cases. Image melding, while taking into account multiple patch transformations, failed to complete the original structure. He and Sun filled in the damaged region based on the



offset statistics. However, it could not find the solution in a perspective space. The results in the fourth and fifth rows show that our method can recover structural consistency. We transform sampled patch in source region into target region using transformation model with a scale variation. The last row illustrates that our algorithm demonstrates outstanding performance in maintaining textural consistency.

### 5.3 Qualitative evaluation

To find a satisfactory completion for the user is the real purpose of image completion. One important test is visual inspection and another one is obtaining quantitative results using peak signal to noise ratio (PSNR). The PSNR comparison of six images in Fig. 5 is shown in Table 1.

We observe that the PSNR value of the proposed algorithm is slightly higher overall than the value of other algorithms. It is easy to know that the images completed by our method are better than the other methods in image consistency and coherence for human eyes. Figure 6 shows the comparisons.

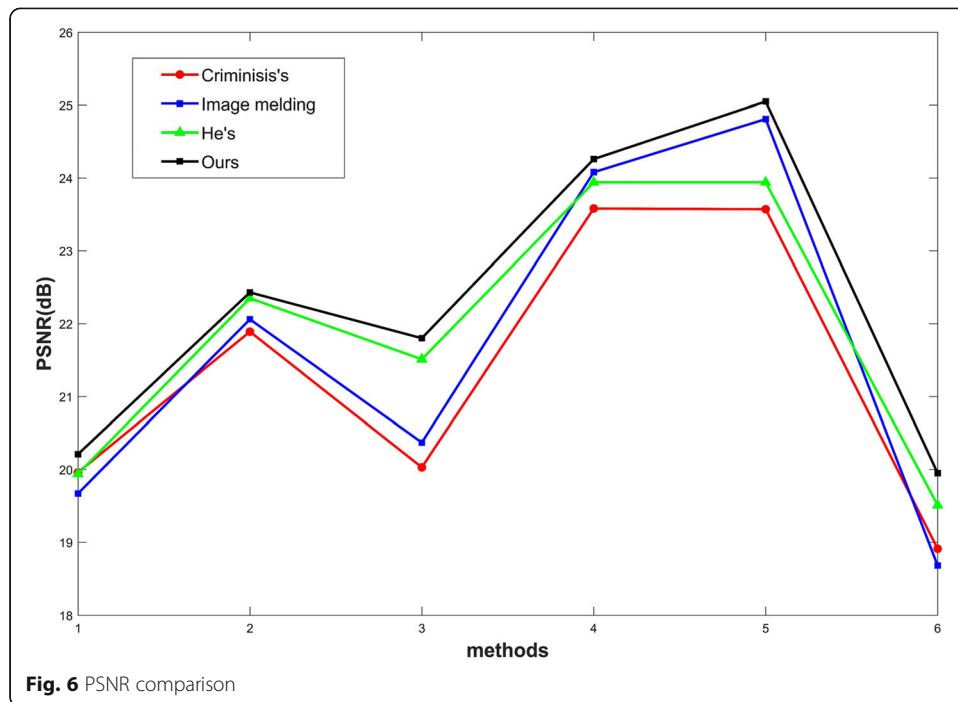
### 5.4 Other results

Object removal is also one of the application occasions of image completion. In order to demonstrate the robustness of our method, we compare our method with current methods in the natural scenes. In these scenes, we cannot acquire a set of plane parameters. Our method can also maintain the consistency of textural structure, and the results satisfy human visual coherence. Figure 7 shows the comparisons with methods from Criminisi [8], Komodakis [10], He [22], and Le Meur [19]. In Table 2, we give the quality scores of the inpainted images, as determined by the technique reported in Ref. [31]. The lower the scores, the better the quality of the image. We can see from the contrast result that Criminisi's method introduces texture in a wrong location. Methods of Komodakis's and Le Meur's can hardly guarantee the structure continuity. He's method achieves more satisfactory inpainting result, while small flaw still exists. Compared with those methods, our method achieves better texture coherence and structure continuity.

In Fig. 8, we compare the proposed algorithm with the method using a deep learning model [23]. The input images are  $128 \times 128$ . We show the results in structural scenes and natural scenes. Compared with deep learning models, our method has better performance in maintaining the structural integrity and the global consistency of texture. The deep learning model repairs the damaged region using a "generate" way. The quality of results relies on numerous training data and excellent network structure. On the contrary, we estimate perspective distortions using a transformation model and

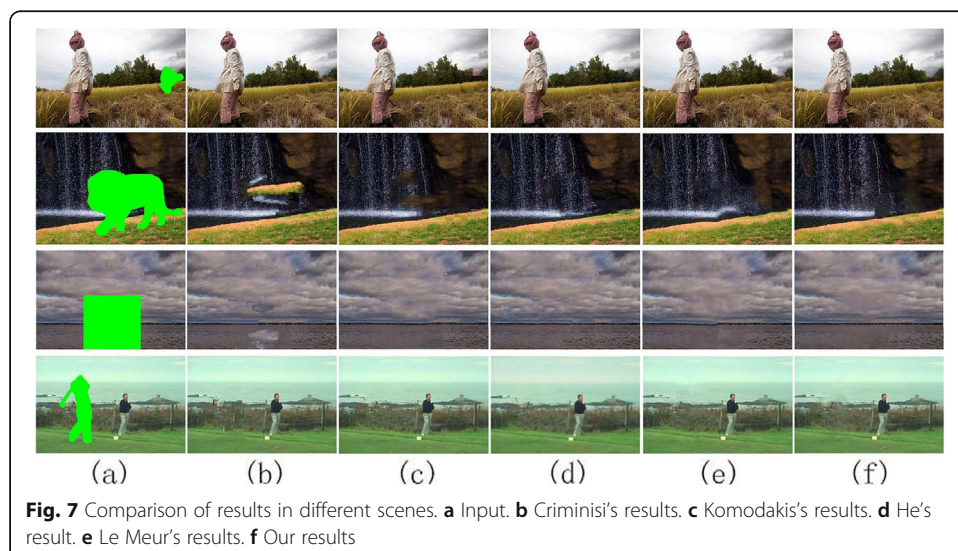
**Table 1** Image completion performance measured in PSNR

Examples	1	2	3	4	5	6	Mean
Criminisi	19.96	21.89	20.83	23.58	23.57	18.91	21.32
Image melding	19.67	22.06	20.37	24.08	24.81	18.68	21.61
He	19.94	22.35	21.51	23.94	23.94	19.51	21.87
Ours	20.21	22.43	21.80	24.26	25.05	19.95	22.28



constrain the completion process using the guidance map. The PSNR (dB) value is shown in Table 3.

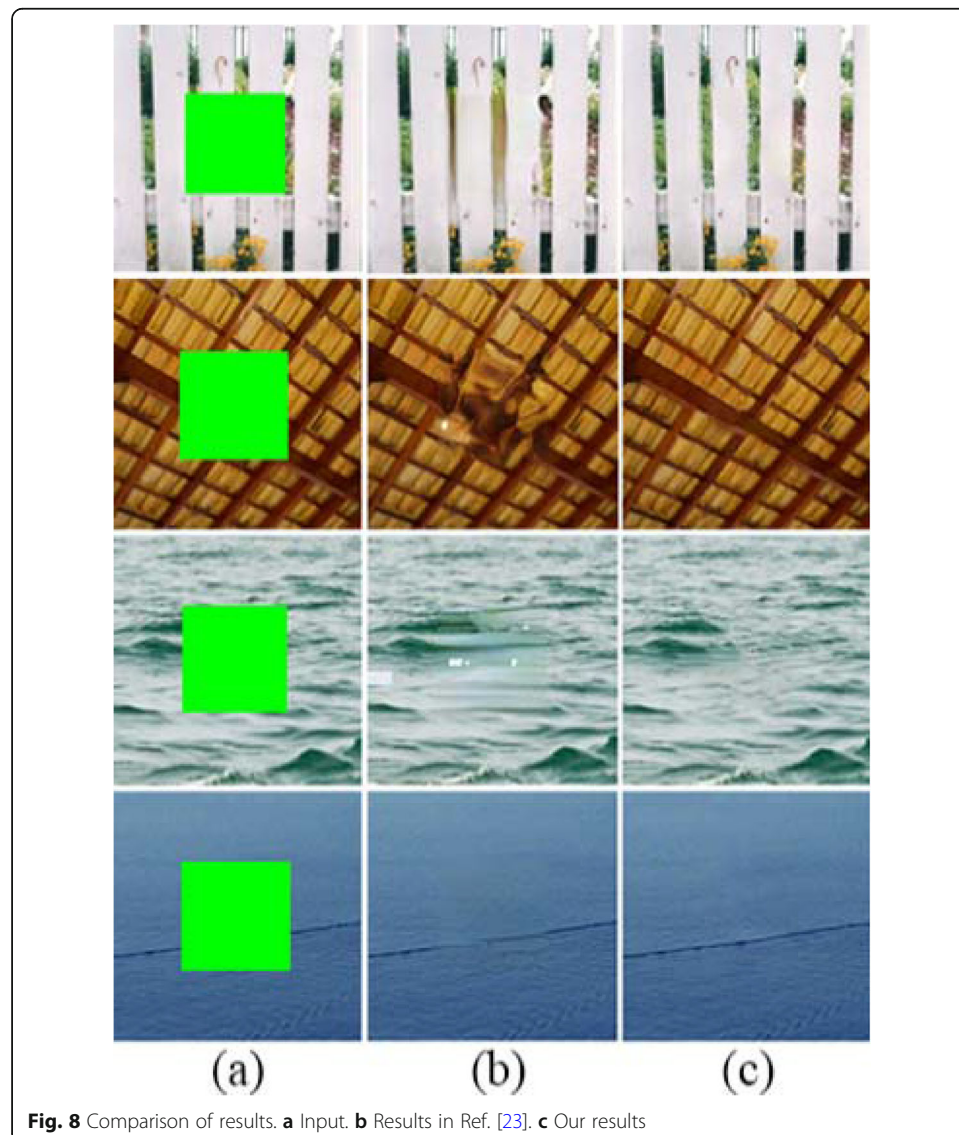
Figure 9 shows the comparison of results by the proposed method and Huang’s method [16]. Form the first row, we can see that our algorithm has better performance when inpainting large damaged region. The second row shows the comparison of results in a perspective scene. Due to the lack of search space and scale constraints, the structure was distortions at the end of the building in Huang’s result. In the third row, we show the comparison in keeping texture continuity. Huang’s method failed to find



**Table 2** The quality scores for Fig. 7

Examples	1	2	3	4	Mean
Criminisi	4.4	4.78	5.95	5.85	5.25
Komodakis	4.14	4.75	5.97	5.5	5.09
He	4.05	4.75	5.88	5.6	5.07
Le Meur	4.18	4.74	6.08	5.6	5.15
Our	3.96	4.71	5.89	5.62	5.05

the demarcation between two kinds of texture. The fourth row demonstrates that our method has a plausible performance in maintaining global texture consistency. We apply a gradient term and a consistency term into our objective function to maintain texture details and encourage sampling patch in adjacent areas. Therefore, the proposed method performs better in both continuity and visual effect. The PSNR (dB) value is shown in Table 4.



**Table 3** The comparisons measured in PSNR for Fig. 8

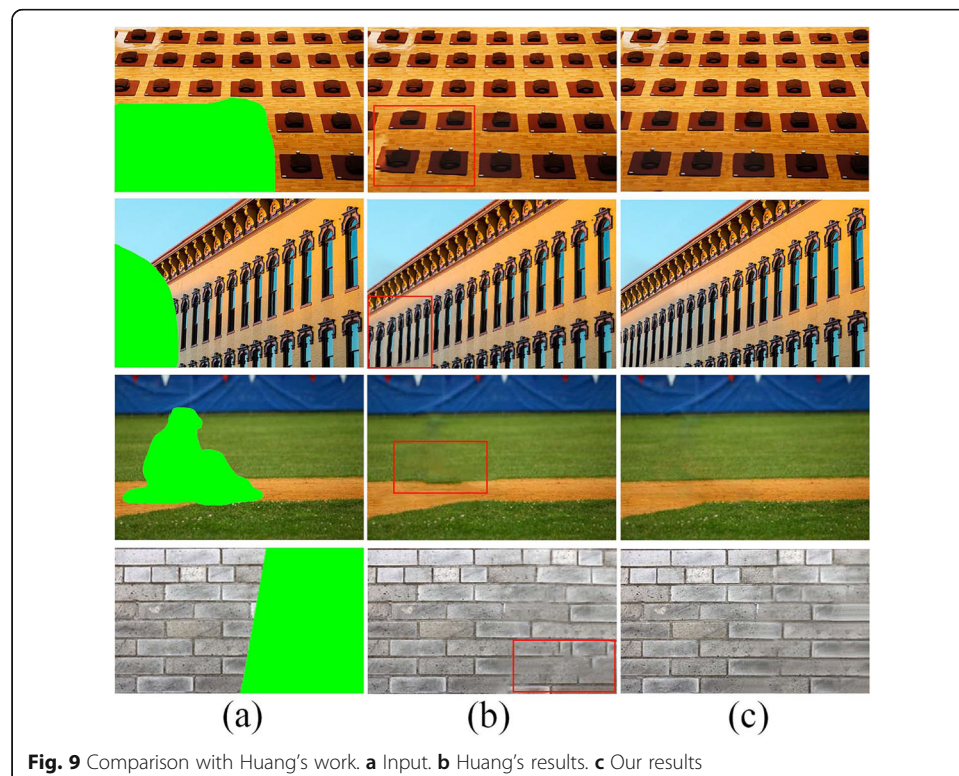
Examples	1	2	3	4	Mean
Ref. [23]	18.83	22.67	23.61	25.27	22.60
Ours	21.73	23.71	23.95	26.94	24.08

### 5.5 Effect of patch size

Figure 10 shows the impact of patch size on the completion results. Our algorithm led to poor performance when using too small patch. Small patch cannot capture enough texture. Similarly, redundant texture was copied when using a too large patch. We apply different patch sizes on an example, as shown in Fig. 10.

### 5.6 Effect of structure guidance

Figures 11 and 12 show the effect of the guidance map and the parameter  $\lambda$ . The guidance map offers significant guidance for the patch searching process. Here, we show the results of our method with different parameter values and comparisons. Figure 11 shows that the structure guidance map can help preserve structure integrity. In Fig. 12, we show the result and effect of parameter  $\lambda$ . We can see that the structure line of the house cannot be repaired reasonably if the value of  $\lambda$  is too small. On the contrary, the structure texture is discontinuous if the value of  $\lambda$  is too large. In our experiments,  $\lambda$  was set to 2.5 and the performance is receivable.

**Fig. 9** Comparison with Huang's work. **a** Input. **b** Huang's results. **c** Our results

**Table 4** The comparisons measured in PSNR for Fig. 9

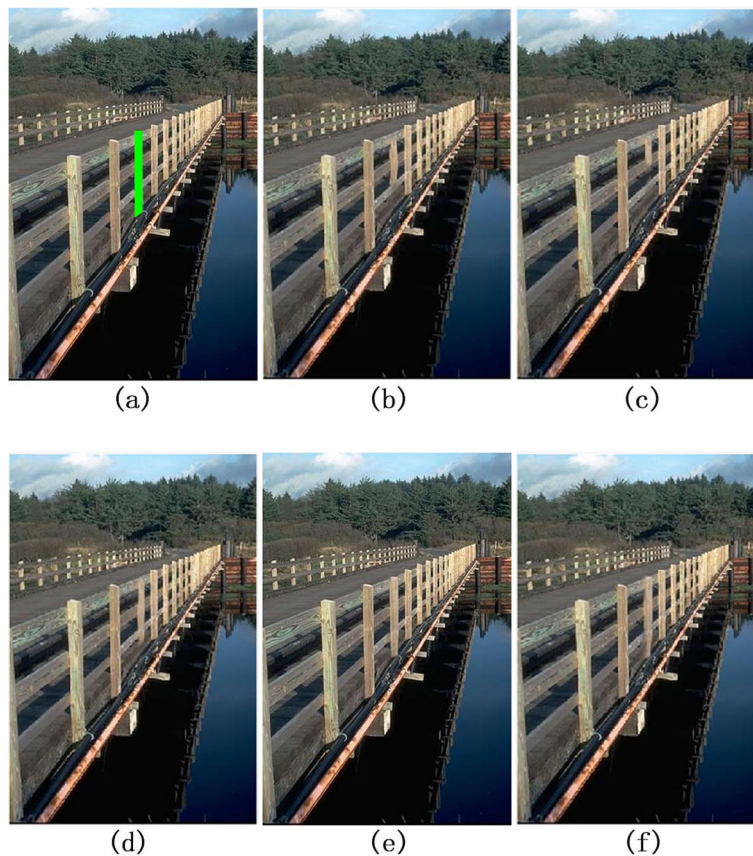
Examples	1	2	3	4	Mean
Huang	18.21	18.34	17.45	18.35	18.09
Ours	19.32	18.82	17.53	19.00	18.67

### 5.7 Effect of gradient term and consistency term

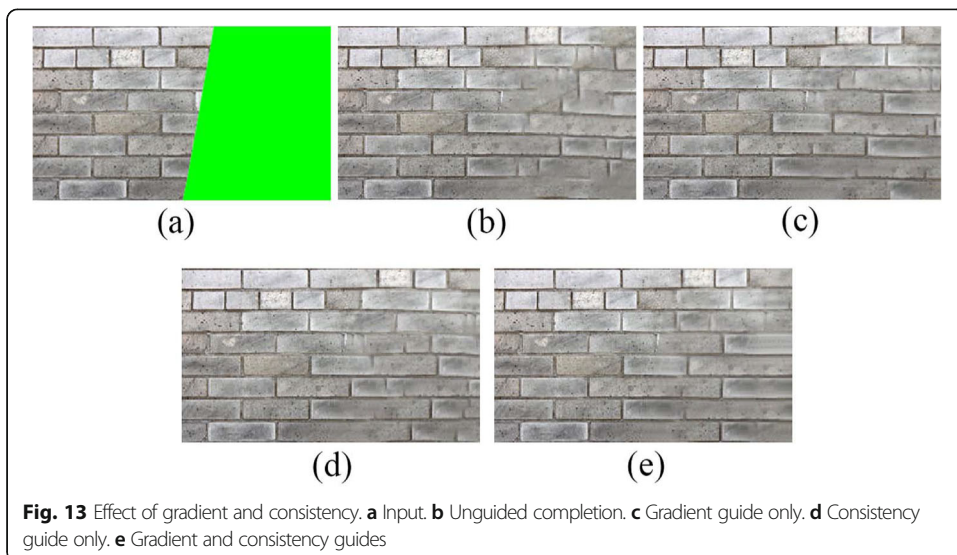
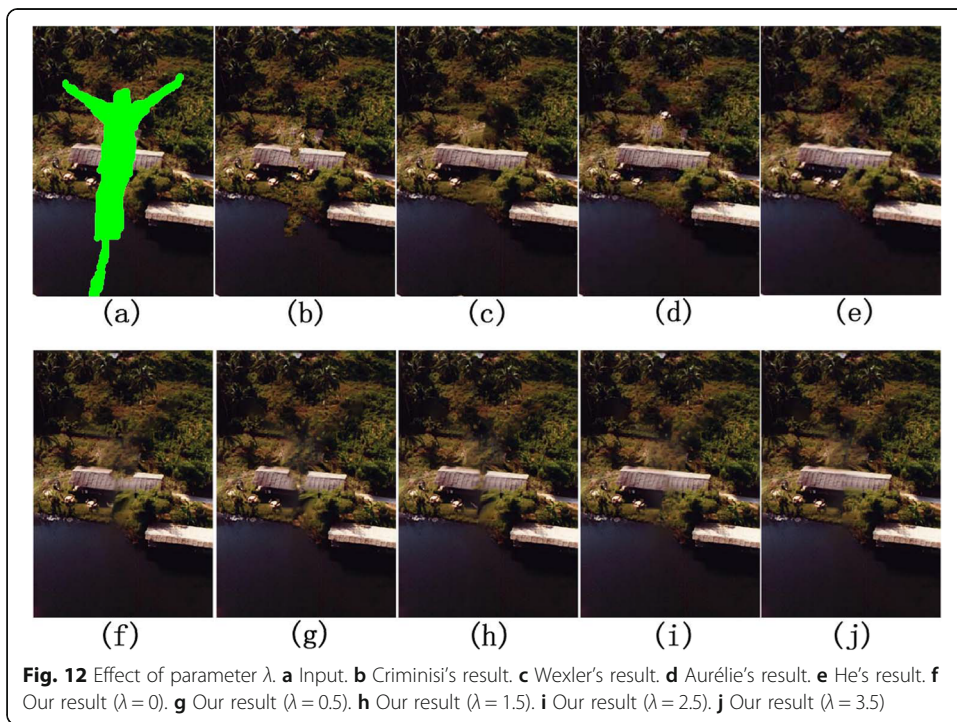
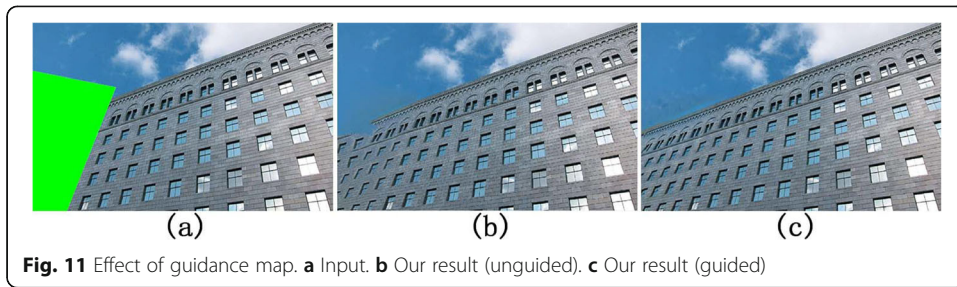
To get some intuition on the importance of the gradient term and consistency term, we illustrate four cases of information usage in Fig. 13. In Fig. 13b, we show the completion result without any guidance. The result is blurry and the structure is wrong. In Fig. 13c, we only use the gradient term in the optimization process. Since the structure information is insufficient, we obtain broken structures. In Fig. 13d, we only use the consistency term. While the completed region has structure information, the texture synthesis has an error in detail. The best result is acquired using both gradient and consistency, as shown in Fig. 13e.

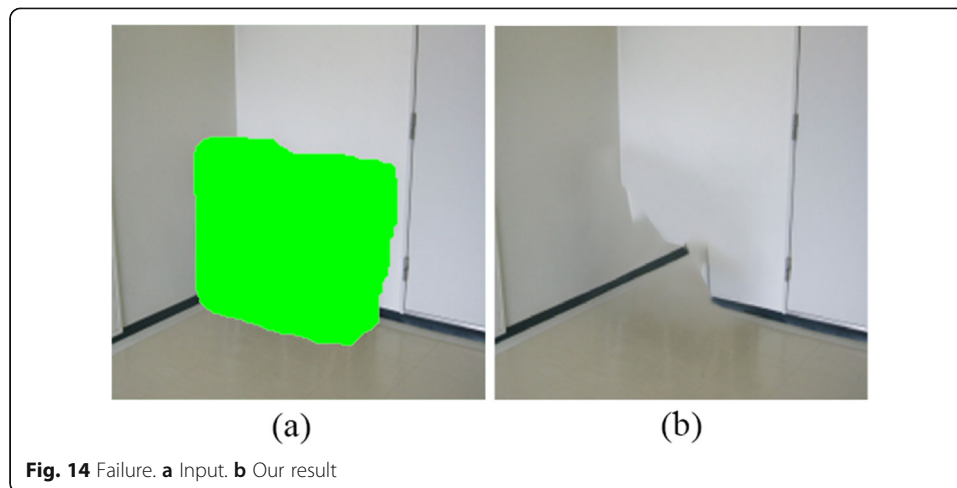
### 5.8 Limitations

It is difficult for our method to handle the texture details if the opposite sides of the hole have textures with very different dominant directions. We fail to complete the structure lines, as shown in Fig. 14. The results may be improved using more



**Fig. 10** Effect of patch size. **a** Input. **b**  $5 \times 5$  patch size. **c**  $7 \times 7$  patch size. **d**  $9 \times 9$  patch size. **e**  $11 \times 11$  patch size. **f**  $13 \times 13$  patch size





sophisticated computer vision methods, which we leave to future work. Running time is also a limitation. Our approach is just a prototype. The time cost of our approach can be further improved using more efficient algorithms.

## 6 Conclusion

We have proposed an improved image completion method using structural constraints. First, we adopted a parameterized transformation model with detected plane parameters to extend the patch search space. Furthermore, we proposed an objective function with two constraints to guide the completion process. These two constraints provided effective guidance when searching for the best matching patches. Finally, we combined the constraints and gradient into a framework that could solve more challenging problems. We implemented our method in many images with various scenes and acquired promising results of visual consistency.

### Abbreviations

MRF: Markov random field; CNN: Convolutional neural network; GLCM: Gray-level co-occurrence matrix; PSNR: Peak signal to noise ratio

### Acknowledgements

Not applicable.

### Authors' contributions

QC and GL conceived and designed the study. QC, GL, QX, and MX performed the experiments. QX and LX offered useful suggestions and helped to modify the manuscript. QC, GL, QX, LX, and MX reviewed and edited the manuscript. All authors read and approved the manuscript.

### Authors' information

Qiaochuan Chen received his BS degree in information security from Hefei University of Technology, China, in 2013, and his MS degree in computer technology from Yunnan University, China, in 2015. Since 2015, he has been working toward his PhD in the College of Electronics and Information Engineering, Tongji University. His current research interests include pattern recognition, image inpainting, and image encryption.

Guangyao Li received his BS and MS degrees from Nanjing University of Aeronautics and Astronautics, China, in 1986 and 1989, respectively. He received his PhD from Nanjing University of Aeronautics and Astronautics in 1997. Now, he is a professor and doctoral supervisor in the College of Electrical and Information Engineering, Tongji University. His primary research interests include image processing and virtual reality.

Qingguo Xiao received his BS degree in communication engineering from Zhejiang Normal University, China, in 2012, and his MS degree in electronics and communication engineering from Ningbo University, China, in 2015. Since 2016, he has been working toward his PhD in the College of Electronics and Information Engineering, Tongji University. His current research interests include pattern recognition, artificial intelligence, and data analysis.



Li Xie received his BS degree in computer science from Zhejiang University of Technology, China, in 2012. Since 2014, he has been working toward his PhD in the College of Electronics and Information Engineering, Tongji University, China. His current research interests include machine learning, pattern recognition, and remote sensing classification. Mang Xiao received his BS degree from Jiangxi Normal University in 2005 and his MS degree in computer science from Nanchang University in 2010. He received his PhD from Tongji University in 2016. Now, he is a lecturer in the School of Computer Science and Information Engineering, Shanghai Institute of Technology. His research interests include image processing and machine learning.

#### Funding

This work was supported by the National Natural Science Foundation of China under grant number 61771346.

#### Availability of data and materials

Data will not be shared; the reason for not sharing the data and materials is that the work submitted for review is not completed. The research is still ongoing, and those data and materials are still required by the author and coauthors for further investigations.

#### Competing interests

The authors declare that they have no competing interests.

#### Author details

<sup>1</sup>College of Electronics and Information Engineering, Tongji University, Shanghai, China. <sup>2</sup>School of Computer Science and Information Engineering, Shanghai Institute of Technology, Shanghai, China.

Received: 4 December 2018 Accepted: 22 September 2020

Published online: 08 October 2020

#### References

1. C. Yan, H. Xie, J. Chen, et al., A fast Uyghur text detector for complex background images. *IEEE Transactions on Multimedia*. **20**, 3389 (2018)
2. C. Yan, L. Li, C. Zhang, et al., Cross-modality bridging and knowledge transferring for image understanding. *IEEE Transactions on Multimedia* (2019)
3. C. Yan, Y. Tu, X. Wang, et al., STAT: spatial-temporal attention mechanism for video captioning. *IEEE Transactions on Multimedia* (2019)
4. M. Bertalmio, G. Sapiro, V. Caselles et al. Image inpainting, in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (ACM Press/Addison-Wesley Publishing Co., 2000), pp. 417.
5. F. Bornemann, T. März, Fast image inpainting based on coherence transport. *Journal of Mathematical Imaging and Vision*. **28**, 259 (2007)
6. D. Tschumperlé, Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's. *Int. J. Comput. Vis.* **68**, 65 (2006)
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* (IEEE, 1999), pp. 1033.
8. A. Criminisi, P. Pérez, K. Toyama, Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. Image Process.* **13**, 1200 (2004)
9. J. Sun, L. Yuan, J. Jia et al., Image completion with structure propagation, in *ACM Trans. Graph.* (ACM, 2005), pp. 861.
10. N. Komodakis, G. Tziritas, Image completion using efficient belief propagation via priority scheduling and dynamic pruning. *IEEE Trans. Image Process.* **16**, 2649 (2007)
11. Y. Wexler, E. Shechtman, and M. Irani, Space-time completion of video. *IEEE Trans. Pattern Anal. Mach. Intell.* **29** (2007).
12. C. Barnes, E. Shechtman, A. Finkelstein, et al., PatchMatch: a randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.* **28**, 24 (2009)
13. A. Mansfield, M. Prasad, C. Rother et al. Transforming image completion, in *BMVC* (2011), pp. 1.
14. S. Darabi, E. Shechtman, C. Barnes et al., Image melding: combining inconsistent images using patch-based synthesis. *ACM Trans. Graph.* **31**, 82:1 (2012).
15. J.-B. Huang, J. Kopf, N. Ahuja et al., Transformation guided image completion, in *Computational Photography (ICCP), 2013 IEEE International Conference on* (IEEE, 2013), pp. 1.
16. J.-B. Huang, S.B. Kang, N. Ahuja, et al., Image completion using planar structure guidance. *ACM Trans. Graph.* **33**, 129 (2014)
17. J.-B. Huang, A. Singh, and N. Ahuja, Single image super-resolution from transformed self-exemplars, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 5197.
18. M. Xiao, G. Li, L. Peng, et al., Completion of images of historical artifacts based on salient shapes. *Optik-International Journal for Light and Electron Optics*. **127**, 396 (2016)
19. O. Le Meur, M. Ebdelli, C. Guillemot, Hierarchical super-resolution-based inpainting. *IEEE Trans. Image Process.* **22**, 3779 (2013)
20. J. Kopf, W. Kienzle, S. Drucker, et al., Quality prediction for image completion. *ACM Trans. Graph.* **31**, 131 (2012)
21. Y. Pritch, E. Kav-Venaki, and S. Peleg, Shift-map image editing, in *Computer Vision, 2009 IEEE 12th International Conference on* (IEEE, 2009), pp. 151.
22. K. He and J. Sun, Statistics of patch offsets for image completion, in *Computer Vision—ECCV 2012* (Springer, 2012), pp. 16.
23. D. Pathak, P. Krahenbuhl, J. Donahue et al., Context encoders: Feature learning by inpainting, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 2536.
24. S. Iizuka, E. Simo-Serra, H. Ishikawa, Globally and locally consistent image completion. *ACM Trans. Graph.* **36**, 107 (2017)
25. M. Xiao, G. Li, Y. Jiang et al., Image completion using belief propagation based on planar priorities. *KSII Transactions on Internet & Information Systems*. **10** (2016).

26. R. Hartley, A. Zisserman, *Multiple view geometry in computer vision* Cambridge University Press (2003)
27. A. Hertzmann, C. E. Jacobs, N. Oliver et al. Image analogies, in Proceedings of the 28th Annual Conference on Computer graphics and interactive techniques (ACM, 2001), pp. 327.
28. S. Zarif, I. Faye, D. Rohaya, Image completion based on statistical texture analysis. *Journal of Electronic Imaging*. **24**, 013032 (2015)
29. O. Chum and J. Matas, Planar affine rectification from change of scale, in Asian Conference on Computer Vision (Springer, 2010), pp. 347.
30. Y. HaCohen, E. Shechtman, D.B. Goldman, et al., Non-rigid dense correspondence with applications for image enhancement. *ACM Trans. Graph.* **30**, 70 (2011)
31. A. Mittal, R. Soundararajan, A.C. Bovik, Making a “completely blind” image quality analyzer. *IEEE Signal Processing Letters*. **20**, 209 (2013)

### **Publisher’s Note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

---

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)

---