An Effective Content-Aware Image Inpainting Method

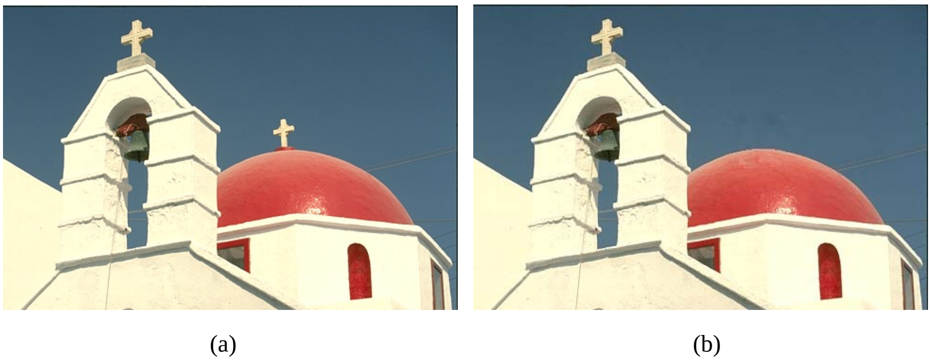
JIUNN-LIN WU* AND I-YING CHOU
Institute of Networking and Multimedia, Dept. of Computer Science and Engineering,
National Chung Hsing University
250, Kuokuang Rd., 402 Taichung, Taiwan
Email: jlwu@cs.nchu.edu.tw

Image inpainting is a technique for restoring damaged old photographs and removing undesired objects from an image. The basic idea behind the technique is to automatically fill in lost or broken parts of an image using information from the surrounding area. The challenge of current inpainting algorithms is to restore both texture and structure characteristic information for large and thick damaged regions. This paper presents a new hybrid image inpainting method based on Bezier curves which combines the exemplar-based inpainting technique and the edge-based image restoration algorithm. For restoring image structures, we first use the segmentation result with iterative Otsu's thresholding to obtain the information of edges. Bezier curves are then used to reconstruct the image skeletons in missing areas. It improves the limitation of the edge-based restoration approach which approximates incoming edges with only lines and circle arcs. The inpainting process is divided into two phases: the first phase restores the image structure by pixel-based interpolation, while the second phase fills holes for preserving texture information with patch-based inpainting method. Experimental results on both synthetic and real images demonstrate the effectiveness of the proposed method. By restoring the curvature structures and textures of the damaged regions, the proposed method achieves content-aware image inpainting.

Keywords: Image inpainting, texture synthesis, exemplar-based method, edge structure reconstruction, pixel-based interpolation, Bezier curves

1. INTRODUCTION

Image inpainting is an important research issue in image restoration. The technique restores the damaged part of an image (called the missing region or the target region) by referring to the information from the undamaged part (called the source region), and bases on certain reasonable rules to make the final restored image look “complete” and “natural”. The applications of the image inpainting technique include removing scratches in a photograph, repairing damaged areas in ancient paintings, removing undesired objects from an image, and more significantly, restoring precious calligraphies and paintings in the digital museum. Recently, digital inpainting also is used to restore an image by removing noise, improving brightness, color and details. The challenge of current inpainting algorithms is to restore both texture and edge characteristic information for large and thick damaged regions. Merging the missing region with its surroundings and preserving both texture and structure information are the research key of image inpainting as shown in Fig.1. If we want to remove the crucifix from the image, how to keep the texture of the sky and the red roof, and rebuild the structure of roof is the main goal of digital inpainting work.



(a) (b)

Fig. 1. Image inpainting. (a) Original image. (b) The result of removing white crucifix from the top of red roof via the proposed image inpainting technique.

Many approaches about image inpainting have been proposed. Roughly speaking, there have been two main types for solving the image inpainting problem. One is the PDE(partial differential equation)-based method, and the other is the texture synthesis approach. The way of PDE-based method is based on the partial differential diffusion equations, and it propagats edge information from outside of the missing region along the isophote directions which are obtained along the inpainting contour by computing the vector perpendicular to the gradient. Therefore this method can preserve all structure information. The main drawback of almost all PDE-based methods is that the final inpainting result may produce blurring artifacts when the missing regions are large and textured. So this type of method best performs those images with pure structure or thin cracks and text.

The second type of approach is based on the texture synthesis, and it overcomes the disadvantage of the PDE-based method. The method fits for the large target region and can preserve linear structure information by adjusting the order of inpainting. However, the structures of almost all images don't only include linear structure information. The drawback of this method is that it couldn't handle curved structures. According to our analysis, we know that it's hard to reconstruct a complex structure and preserve texture information for large and thick damaged regions.

In this paper, a new hybrid inpainting method is presented. The main idea is that we first reconstruct the edge structures by using Bezier curves and then fill the unknown region by copying content from the undamaged part of image by using the patch-based inpainting technique. Many examples included artificial and real images have shown the effectiveness of the proposed method. The rest of this paper is organized as follows: Section 2 is a brief survey of related work on variable inpainting methods. Section 3 describes the proposed Bezier curves based inpainting algorithm in detail. Experiments on both synthetic and real images are presented in section 4. Finally we conclude this paper in section 5.

2. RELATED WORKS

The literature has recently developed various image inpainting approaches, and they broadly are categorized into several types as follows. Bertalmio *et al.* [1] first introduced the mathematically computational inpainting method, which is based on partial differential equations (PDE). This method filled the missing region in an image by using the partial differential diffusion equation to propagate information from surroundings of the missing region along the isophote direction which is perpendicular to the gradient vector of the restored region contour, and it intended to propagate information while preserving edges. Based on the idea of Bertalmio *et al.* [1], some similar methods were proposed. For example, Chan and Shen developed general mathematical models [2] and used curvature driven diffusions (CDD) [3] for image inpainting. Usually PDE-based algorithms are designed to remove occlusions, such as text, subtitles and stamps from images. However, the PDE-based method implicitly assumes that the content of the missing region is smooth and non-textured. So, the final inpainting result may over-smooth the image and produces blurring artifacts when the missing region is large or the content is textured.

For solving the blurring problem when filling larger regions and disturbing artifacts if the inpainting domain is surrounded by textured regions, texture-synthesis based inpainting schemes were proposed. Those methods attempted to replicate texture in an image to fill the target region [4]-[11], therefore the texture information is preserved. Texture synthesis based approaches are classified into pixel-based sampling [4]-[7] and patch-based sampling [8]-[11] according to the sample texture size. They performed well in the case of the missing regions with "pure" textures. A major drawback of these related approaches is that their greedy method of filling the image often leads to visual inconsistencies. That means color, texture or structure are not continuous. And texture-synthesis based methods usually do not consider the image structure information around the target region. The filling order adopts the traditional concentric-layer filling approach or from left to right and then from up to down, so this kind of methods can only be applied to the images with pure textural backgrounds. If the damaged image consists of linear structures and composite textures, the texture-synthesis based inpainting method cannot preserve the linear structure when inpainting.

Therefore, some researchers have recently proposed algorithms which combine the advantages of partial differential equations and texture synthesis. These methods simultaneously propagate texture and structure information. Bertalmio *et al.* [12] proposed an algorithm which decomposes the image into structure and texture components, and restores them by PDE and texture synthesis, respectively. The final result is the sum of two restored (processed) components. This method is not suitable for repairing large and thick damaged regions since the PDE approach often produces the blur artifacts.

Drori *et al.* [13] proposed the fragment-based image completion. Their method is to iteratively approximate the unknown parts and select adaptive image fragment to restore the image. The order of inpainting is decided based on the confidence map. Beside, the size of the fragment is decided according to the smooth degree of the image. Although its result is good, but the speed is very slow.

Criminisi *et al.* [14] proposed the exemplar-based image inpainting algorithm to preserve linear structure by reordering the synthesizing process according to the priority. The priority of synthesizing order is decided by the product of two terms: the pixel number in the source region and the dot product between the isophote direction and the normal vector of contour. After deciding the priority, the exemplar-based image inpainting algorithm uses patch-based texture synthesis to replicate the texture in an image to fill the target region and searches for the most similar texture by the sum of squared differences (SDD). This approach inherits the texture synthesis feature in producing the texture characteristic and propagates the linear structure into the missing region. Its advantage lies in constructing both texture and structure characteristic information for large and thick damaged regions. Some methods based on the exemplar-based inpainting algorithm are reported recently[19, 20] However, these methods has limitations, for example, it can't handle other types of structures such as curves and the ambiguities in which the missing region covers the intersection of two regions.

Sun *et al*. [15, 21] introduced a structure propagation approach to image inpainting for other types of structures. In this method, the users are required to specify manually the important missing structures by extending a few curves or line segments from a known region to the unknown. Image patches are then synthesized along the user-specified curves using patches selected around curves in the known region. After completing the structure propagation, the remaining unknown regions are filled using the conventional patch-based texture synthesis method. The drawback of the above scheme is that finding image structure is not automatic. Shen *et al*. used the gradient maps in the removed area to completed through a patch-based filling algorithm, the image is reconstructed from the gradient maps by solving a Poisson equation [22]. Its result provides better details than the conventional exemplar-based method. Komodakis and Tziritas [23] proposed a optimization scheme, priority belief propagation(BF), to improve the occurrence of visually inconsistent results. However, they did not deal with the problem of edge structure reconstruction.

Rares *et al*. [16] presented an edge-based image restoration algorithm, in which a novel edge-pairs matching method is proposed to reconstructs the edge structure in the missing area. Several important and reasonable rules for a matched edge pair is considered, for example, the intensity and colors of objects separated by an edge should be consistent, the estimation of how well one edge continues into another one, and the direction of edges with respect to each other. Afterward, the reconstructed edges guide the following inpainting process. Before reconstructing the edge structure, they employed watershed algorithm to segment the image. The edges in the known area are detected, and edge features are then extracted for edge-pairs matching algorithm. Then the authors utilized the above mentioned rules and the grouping procedure to restore the edge structure in missing area. After finding the edge structure, they used the circle function to find the edge location of every pixel and adopted pixel-based interpolation to reconstruct the missing region. A main limitation of the above method is that it can not handle the case of the image with curvature structure. The pixel-based method is effective for small image gaps like scratches in a photograph, however, the reconstructed image becomes unclear when the missing region is large. Using the pixel-based method to reconstruct the missing region can not preserve the texture information of an image compared to using the patch-based method. Moreover, the edge-based inpainting method is time-consuming.

To reconstruct a complex structure and preserve texture information of an image without having to specify the curve, this paper proposes a new image inpainting method, which refers to the edge-based image restoration algorithm [16] and improves its drawback. This study adopts ideas from the previous analysis to modify the edge-based image restoration algorithm, the contributions of the proposed method include: (1) In order to speed up the image segmentation process and improve the over-segmentation problem caused by the watershed algorithm, an iterative image thresholding [17] is used for segmentation, which is equivalent to the fast multi-threshold Otsu's algorithm. (2) A Bezier-curve based method is proposed to depict the curves which connect the matched edges. Bezier curves make the connected edge more natural and reconstruct the image structure including linear and quadratic curves, such as arc, part of an ellipse. (3) A hybrid technique is used for inpainting process, which combines with the pixel-based interpolation and patch-based image inpainting algorithm [14]. The pixel-based method interpolates the edge structure precisely, and the exemplar-based method preserves texture information, and fits to repair large damaged regions.

3. THE PROPOSED METHOD

Figure 2 illustrates the flowchart of the proposed method. The main characteristics of our method are the segmentation approach in step1 and the inpainting process as shown in step4 and step5.

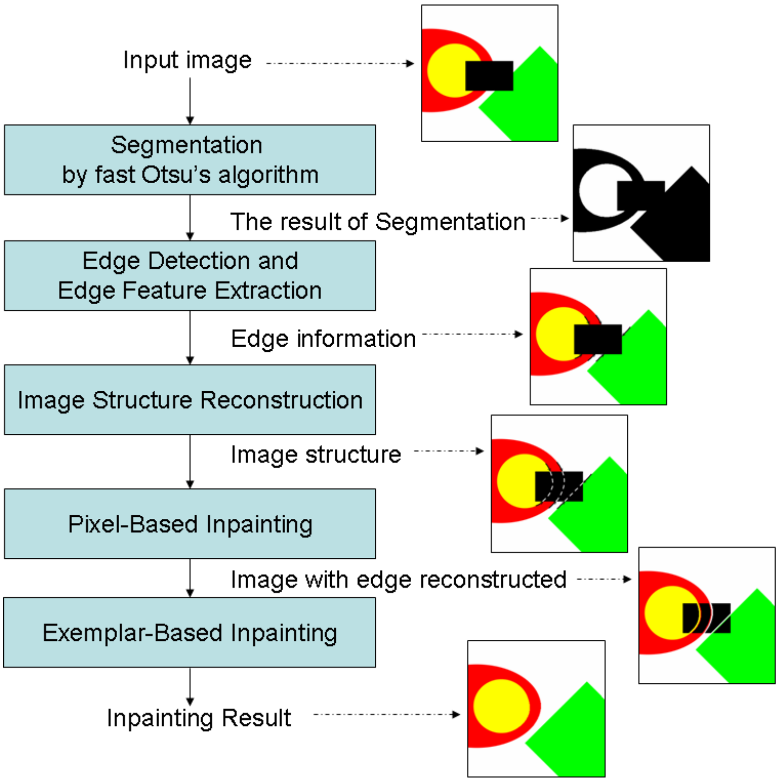

Input image
Segmentation by fast Otsu's algorithm
The result of Segmentation
Edge Detection and Edge Feature Extraction
Edge information
Image Structure Reconstruction
Image structure
Pixel-Based Inpainting
Image with edge reconstructed
Exemplar-Based Inpainting
Inpainting Result

Fig. 2. The flowchart of the proposed image inpainting method.

Our method consists of five main steps:

Step1: The target of our proposed method is reconstructing the edges with both linear and curved structures. Matching the edge pairs are based on the considerations including the angular consistency, sequentiality and proximity of edges. For obtaining the edge information and features, the first step of the proposed method is image segmentation. In this work, an iterative image thresholding method [17] is used, that is a fast version of multi-threshold Otsu's algorithm. The way of this method is that find optimal threshold from image histograms and then applies the optimal threshold to segment images. Besides, the computational complexity of this method is linear. It does not suffer from the over-segmentation problem and the computation time is fast compared to other segmentation methods, such as watershed algorithm. The detail is presented in section 3.1.

Step2: After image segmentation step, edges are detected based on the contours of the segments that resulting from iterative image thresholding [17] in the above step. The edge features are then extracted for each edge, they include the intensity on both sides of the edge, and the magnitude of the local gradient along the edge. This information is used for the next steps.

Step3: Our target is finding the image structures which are lost in the missing area. Edge matching is performed according to the following rules:

(1) The two luminance values on both sides of each edge in the edge couple should be similar.

(2) The magnitude of the local gradients of both edges should be similar.

(3) The degree to which the edge couple fits a common curve is high, in other words, the curvature of two edges in an edge couple should be similar.

(4) Two edges in an edge couple can not be positive edges or negative edges simultaneously, for example, as illustrated in Fig. 3, edge a and edge d are positive edges; on the contrary, edge b and edge c are negative edges. The rule of determining the positive edges or negative edges is to consider the direction of the edge. The direction of the edge means the tail of an edge points to the head of an edge. The head of an edge means the end which is near to the damaged regions. On the contrary, the tail of an edge means the end which is far to the damaged regions.

(5) The angle between the heads of the two edges in a couple should be smaller than the angle between the tails of the two edges in a couple.

(6) The number of crossing couples in the best configuration should be minimal.

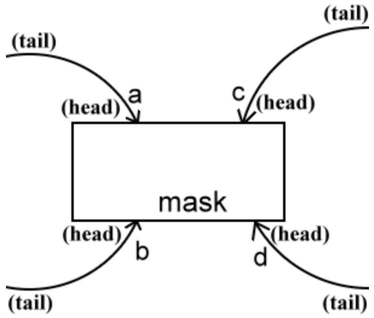


Fig. 3. The illustration of the positive edges or negative edges according the direction of edges.

Based on the aforementioned rules for edge pairs, the edge grouping procedure decides the matched edges with best configuration. We refer to the edge-based image restoration algorithm [16] to find the matching edges. The method includes four steps.

Step(i): Calculate the feature distance d_{kl} between all possible edge pairs (e_k, e_l) by the following formula[16]:

$$d_{kl} = \sqrt{\frac{\left(\beta_{kl}^{C}\left(\lambda 1_{kl}^{C} - \lambda 2_{kl}^{T}\right)^2 + \beta_{kl}^{T}\left(\lambda 1_{kl}^{T} - \lambda 2_{kl}^{C}\right)^2 + \beta_{kl}^{C}\beta_{kl}^{T}\left(\gamma 1_{kl} - \gamma 2_{kl}\right)^2 + \omega_{kl}^{2}\right)}{\beta_{kl}^{C} + \beta_{kl}^{T} + \beta_{kl}^{C}\beta_{kl}^{T} + 1}} \quad (1)$$

where $\lambda 1_{kl}^{C}$, $\lambda 1_{kl}^{T}$, $\lambda 2_{kl}^{C}$ and $\lambda 2_{kl}^{T}$ represent the intensity value on the two side of the corresponding edges e_k and e_l respectively. $\gamma 1_{kl}$ and $\gamma 2_{kl}$ are the magnitude of gradient along edge e_k and e_l respectively. $\beta_{kl}^{C} \in \{0,1\}$ and $\beta_{kl}^{T} \in \{0,1\}$ represent whether the next edge on the two side of edge e_k belongs to a edge pair in the same group($\beta = 1$) or not($\beta = 0$). ω_{kl} is the cost function of fitting a circle to edge pair (e_k, e_l).

$$\omega_{kl} = 1 - \delta_{kl} \times \varphi_{kl} \times \theta_{kl} \quad (2)$$

where φ_{kl} is the degree of angular consistency. θ_{kl} calculates the aperture quality which means the angle between the heads of the two edges in a edge pair and the angle between the tails of the two edges in a edge pair. Finally, δ_{kl} is the spatial deviation of the edge pair from the fitted circle, which is obtained by:

$$\delta_{kl} = \frac{K^{\delta}}{\hat{m}\left(\left|d\left(\left[e_{kl}^{1}, e_{kl}^{2}\right], x_{kl}^{\Theta}\right) - r_{kl}^{\Theta}\right|\right) + K^{\delta}} \quad (3)$$

where e_{kl}^{1} and e_{kl}^{2} are edge e_k and edge e_l. x_{kl}^{Θ} and r_{kl}^{Θ} are the center and the radius of a circle. $d(\ )$ is the euclidean distance, and $\hat{m}$ is the median operation. $K^{\delta} = 2.16$ is a constant for normalization. The value for measuring the angular consistency, φ_{kl}, is defined by [16]:

$$\varphi_{kl} = \frac{\left|\Delta^{*}\alpha_k - \Delta^{*}\alpha_l\right|}{\left|\Delta^{*}\alpha_k\right| + \left|\Delta^{*}\alpha_l\right| + \varepsilon^{\varphi}} \quad (4)$$

where $\Delta^{*}\alpha_{k}$ is the angle which is formed by edge e_{k} . $\Delta^{*}\alpha_{l}$ is the angle which is formed by edge. Finally θ_{kl} is used for measuring the aperture quality[16], which is obtained by

$$\theta_{kl} = \begin{cases} \sqrt{1-\dfrac{\left|\tilde{\Delta}\alpha_{kl}^{h}\right|}{2\pi}}, \text{ if } \left|\tilde{\Delta}\alpha_{kl}^{h}\right| < \left|\tilde{\Delta}\alpha_{kl}^{t}\right| \\ \sqrt{\dfrac{\left|\tilde{\Delta}\alpha_{kl}^{h}\right|}{2\pi}}, \qquad \text{otherwise} \end{cases} \tag{5}$$

where $\tilde{\Delta}\alpha_{kl}^{h}$ is the angle between the heads of the two edges in a edge pair. $\tilde{\Delta}\alpha_{kl}^{t}$ is the angle between the tails of the two edges in a edge pair.

Step(ii): After calculating the feature distance, it determines the potential of two edges to make a couple by the following formula.

$$\pi_{kl} = \pi_{lk} = \begin{cases} TRUE, \text{if } d_{kl} < \min\left(2\times\min_{k} d_{kl}, 2\times\min_{l} d_{kl}, \sqrt{0.1}\right) \\ FALSE, \qquad \text{otherwise} \end{cases} \tag{6}$$

where $\min_{k} d_{kl}$ and $\min_{l} d_{kl}$ represent the minimum of all feature distances including edges e_{k} and e_{l} respectively.

Step(iii): This stage implements the edge grouping procedure. Only the edge pairs which are true need to execute this procedure. It consists of the following substeps [16]:

Step3.1: Initialize group G_{1} with couple $q_{j} = (e_{k}, e_{l})$ and remove it from Q_{i} .

Setp3.2: Search the next edge e_{k}^{next} clockwise form e_{k} and e_{l}^{next} anti-clockwise from e_{l} , as shown in Fig. 4(a). If e_{k}^{next} and e_{l}^{next} belong to one of the possible couples in Q_{i} and the couple $\left(e_{k}^{next}, e_{l}^{next}\right)$ don't intersect any couple from the current group G_{1} , we add couple $\left(e_{k}^{next}, e_{l}^{next}\right)$ to the current group G_{1} and remove it from Q_{i} .

Step3.3: Repeat substep3.2, but the direction is opposite. Look for the next edge e_{k}^{next} anti-clockwise form e_{k} and e_{l}^{next} clockwise from e_{l} , as shown in Fig. 4(b).

Step3.4: If $Q_{i} \neq \phi$ and it has no more couples can be added to the current group G_{1} , we initialize another group with a couple from Q_{i} and repeat the substep3.1-3.3 for the new group.

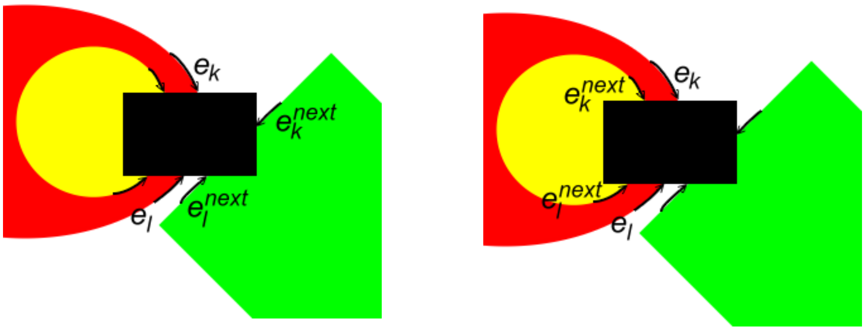
Step3.5: Compute the cost c^{cfg} of the current configuration of groups by:

$$c^{cfg}(Z_{h}) = \frac{4\times c^{loc}(Z_{h}) + c^{\text{glb}}(Z_{h})}{5} \tag{7}$$

where c^{loc} is the matching degree of the couples. c^{glb} is the crossing degree of the couples.

Step(iv): Due to the set Q_{i} doesn't include all possible couples, we can get some possible configurations for the matched edge-pairs. The best configuration Z has the minimum cost.

$$Z = Z_{m}, m = \arg\min_{h} c^{cfg}(Z_{h}) \tag{8}$$



(a) (b)

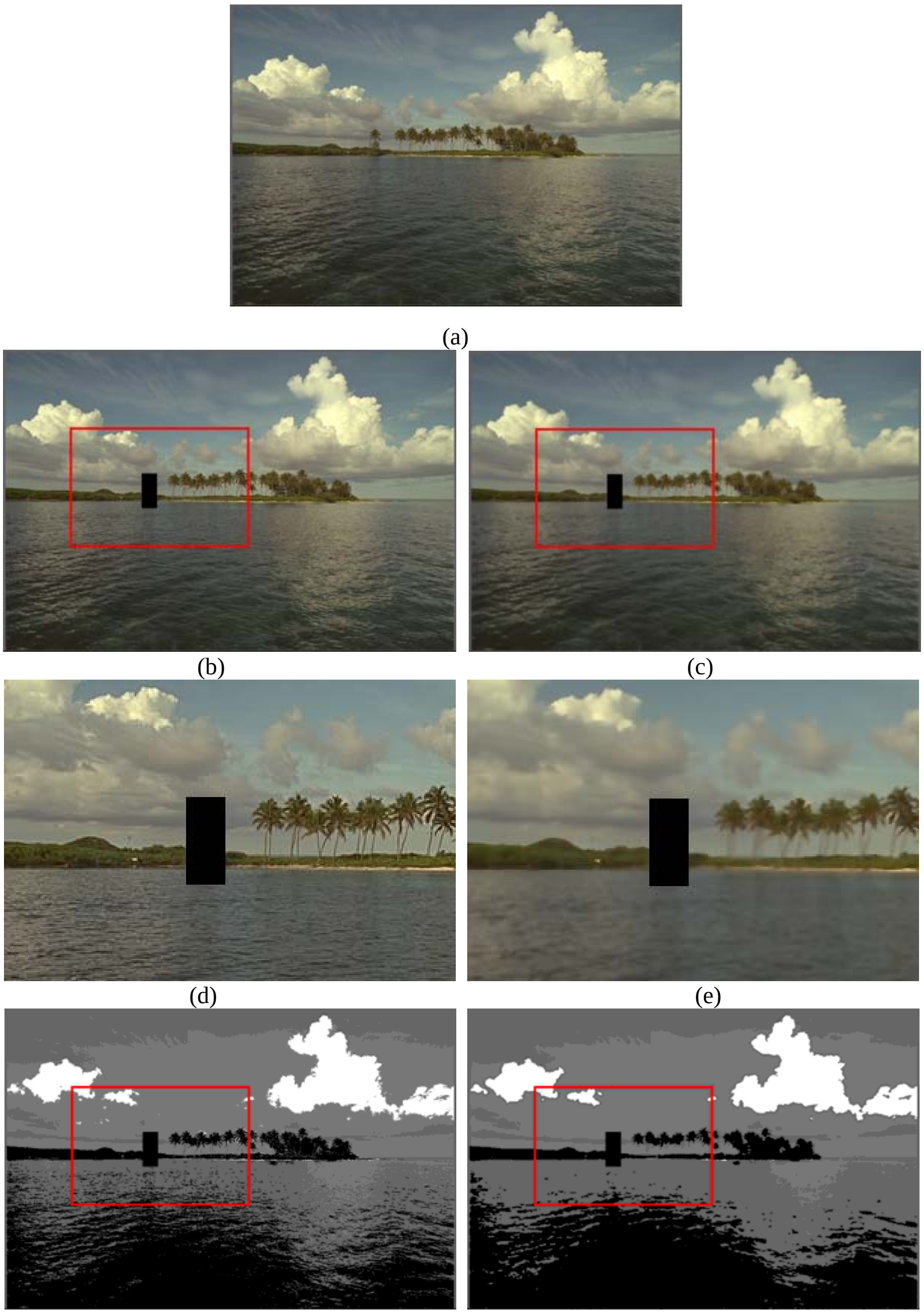
Fig. 4. The illustration of finding the next edges.

Step4: After finding the best configuration by the aforementioned rules and the grouping procedure, we propose to use Bezier curves to depict the curvature skeleton which connect the matched edges and the pixel-based interpolation method is then employed to decide the intensity of the damaged pixels. The detail is presented in section 3.2.

Step5: After reconstructing the image structure in the missing region, a hybrid inpainting method is proposed to restore the remaining parts of the damaged images. The whole inpainting process includes two stages: pixel-based inpainting and exemplar-based inpainting. We employ pixel-based interpolation for building the image skeletons, and perform the exemplar-based image inpainting algorithm to restore the rest of the missing region. The benefit of this method is that it can preserve the information of texture and curved structures. The inpainting time is shorter than the typical pixel-based scheme. The details are described in section 3.3 and 3.4.

3.1 Segmentation by fast multilevel Otsu's algorithm

In our method, the multilevel thresholding algorithm [17] is used for image segmentation. By increasing the number of levels, the iterative multilevel thresholding algorithm [17] can get good segmentation result even if the image content is complicated. When the texture of the damaged images is complex, the segmentation result by the iterative image thresholding algorithm probably produces many broken regions that actually belong to the same region. For reducing the number of broken regions and making the edges between the different textures more apparent, we use an edge-preserving filter, called bilateral filter, to smooth the damaged images before segmentation. The bilateral filter avoids the same texture is distributed into many regions and relieves the over-segmentation problem. Figs. 5(f) and (g) are the segmentation results of the damaged image and the smoothed image which has been processed by the bilateral filter respectively. It is obvious that the number of broken regions in Fig. 5(g) is less than that in Fig. 5(f), the problem of over-segmentation solved. By using the bilateral filter as preprocessing step and the iterative multilevel image thresholding for image segmentation, only the edges with large intensity change are detected, they are exactly the skeletons in image.



(a)

(b)

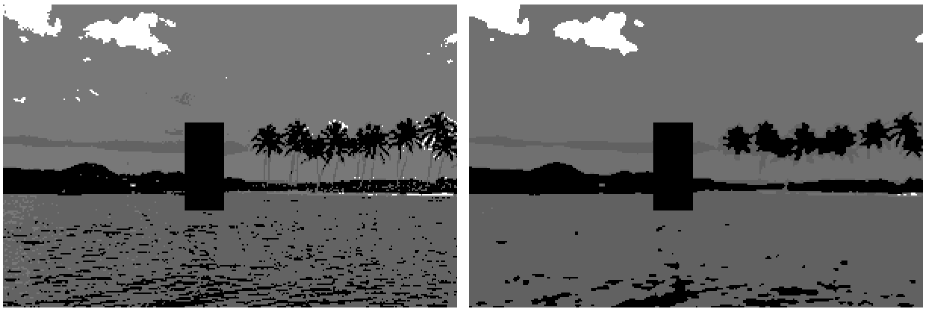
(c)

(d)

(e)

(f)

(g)



(h) (i)

Fig. 5. The result of the segmentation. (a) Original image. (b) The damaged image. (c) The smoothed image by bilateral filering. (d) Zoom-in on the areas of interest in (b). (e) Zoom-in on the areas of interest in (c). (f) The segmentation result of the original damaged image. (g) The segmentation result of the smoothed damaged image. (h) Zoom-in on the areas of interest in (f). (i) Zoom-in on the areas of interest in (g).

3.2 Edge reconstruction using Bezier curves

Bezier curves are widely used in computer graphics to model smooth curves. To describe every kind of curves, we propose to connect the matched edges by Bezier curves. Our inpainting process employs linear and quadratic Bezier curves to build the skeleton of the missing areas. The advantage of Bezier curves is that they can depict complex curves if enough control points are given.

The quadratic Bezier curve is computed by three control points (P_0, P_1, P_2). The formula is defined as follows:

$$B(t) = (1-t)^2 P_1 + 2t(1-t)P_0 + t^2 P_2 \qquad t \in [0,1] \tag{9}$$

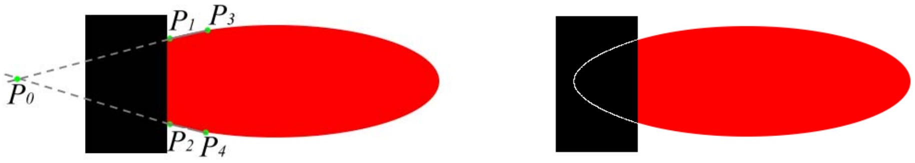
where P_1 and P_2 are the first pixel of the head of the matched edges respectively, P_0 is the intersection point of two edges (i.e. the matched edges), and t is a number varying from zero and one, as shown in Fig. 6(a). In Fig. 6(a), P_3 and P_4 are the tails of two matched edges respectively. P_0 is the intersection of two corresponding line equations formed by two point-pairs (P_1, P_3) and (P_2, P_4) respectively. That is, the coordinate of the point P_0, (x_0, y_0), is calculated by the following formula:

$$\begin{cases} x_0 = \begin{bmatrix} (x_3-x_1)\times(x_4-x_2)\times(x_2-x_1)+ \\ (y_3-y_1)\times(x_4-x_2)\times x_1 - \\ (y_4-y_2)\times(x_3-x_1)\times x_2 \end{bmatrix} / d \\ y_0 = \begin{bmatrix} (y_3-y_1)\times(y_4-y_2)\times(x_2-x_1)+ \\ (x_3-x_1)\times(y_4-y_2)\times y_1 - \\ (x_4-x_2)\times(y_3-y_1)\times y_2 \end{bmatrix} / (-d) \end{cases} \tag{10}$$

where (x_1, y_1), (x_2, y_2), (x_3, y_3) and (x_4, y_4) are the positions of the points P_1 、 P_2 、 P_3 and P_4 respectively. d is the slope which is yielded by.

$$d = (y_3 - y_1)\times(x_4 - x_2) - (y_4 - y_2)\times(x_3 - x_1) \qquad (11)$$

Fig. 6(b) shows the result of using the quadratic Bezier curve to reconstruct an ellipse-shaped structure.



(a)

(b)

Fig. 6. The quadratic Bezier curve. (a) The control points of quadratic Bezier curve. (b) The edge is depicted by quadratic Bezier curve.

If the matched edges can not find the center of the fitting circle (i.e. parallel lines), we employ linear Bezier curves to connect the matched edges, defined as follows:

$$B(t) = (1-t)P_1 + tP_2 \qquad t \in [0,1] \qquad (12)$$

where P_1 and P_2 are the first pixel of the head of the matched edges respectively, and t is a number varying from zero and one. In Fig. 7, the blue nodes are the control points used to construct linear Bezier curves and the gray dotted line is the connected line computed by Eq. (12).

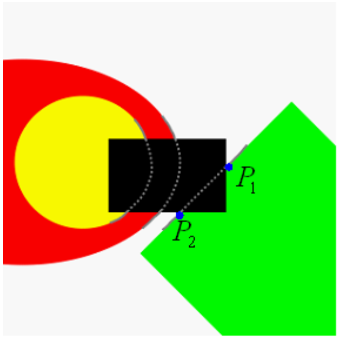


Fig. 7. The diagram of constructing edge by linear Bezier curve.

3.3 Pixel-based inpainting

After finding the best configuration and the position of curves (i.e. using the formula of linear and quadratic Bezier curves), we adopt pixel-based inpainting method to repair the matched edge and find the intensity of pixels. In Fig. 8, the intensity of the pixel P is decided by the pixel P_1 and P_2. The intensity λ^P of the pixel P is defined as follows:

$$\lambda^P = \frac{\omega_1 \lambda^{P_1} + \omega_2 \lambda^{P_2}}{\omega_1 + \omega_2} \tag{13}$$

where λ^{P_1} and λ^{P_2} represent the intensity of pixel P_1 and P_2 respectively, and $\omega_i, i \in \{1,2\}$ is the distance from P to P_i.

$$\omega_i = 1/d\left(x^{P_i}, x^P\right) \tag{14}$$

where x^{P_i} represent the position of the pixel P_i, $d(\ )$ represent the Euclidean distance between the two pixels.

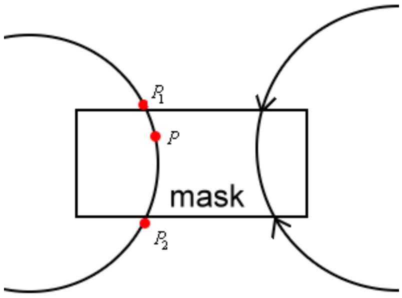


Fig. 8. Using pixel-based inpainting method to draw the skeleton.

3.4 Exemplar-based inpainting

After repairing the edge intensity in the missing region, we apply the exemplar-based image inpainting algorithm [14] to restore the rest of the missing region. The exemplar-based image inpainting algorithm determines the inpainting order by the strength of isophotes and the amount of reliable information. In order to provide more reliable pixels surrounding the restored edges, we widen the edges obtained in the previous subsection. The skeletons are widened by using Eq. (13), the corresponding schematic illustration is depicted in Fig. 9. The intensity of the pixel B is decided by the pixels B_1 and B_2 , while the intensity of the pixel G is computed from the gray levels of the pixels G_1 and G_2 .Fig. 10(a) shows the repaired edge image whose width is only one-pixel. Fig. 10(b) shows the result after widening skeletons, in which the edges are five pixels wide.

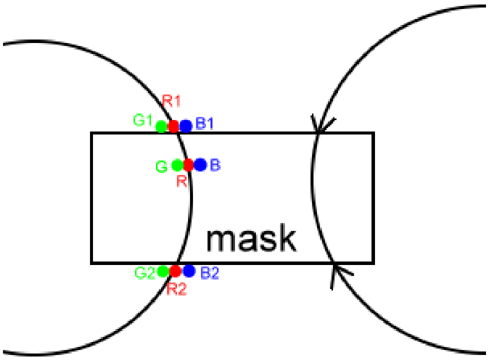


Fig. 9. The illustration of widening the skeletons in the missing area.

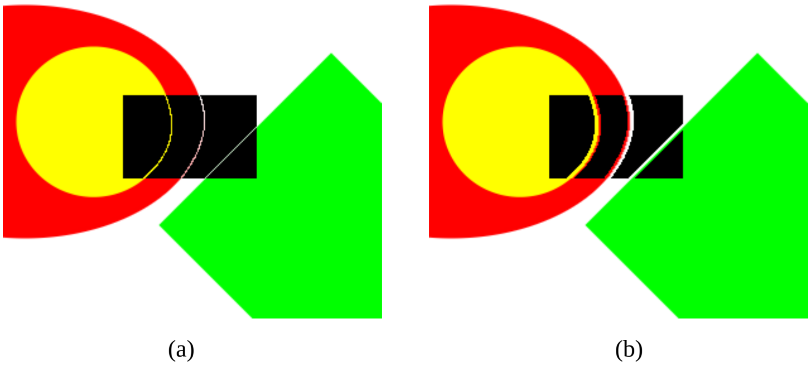


Fig. 10. Example of skeleton restoration via pixel-based inpainting method. (a) The restored one-pixel-wide edges. (b) The result after widening the skeletons.

Figure 11 shows the final result by using the exemplar-based image inpainting algorithm to repair the image shown in Fig. 10(b). It is obvious that the proposed hybrid inpainting method not only preserves the curvature structures but also reconstructs the complex textures in the missing area.

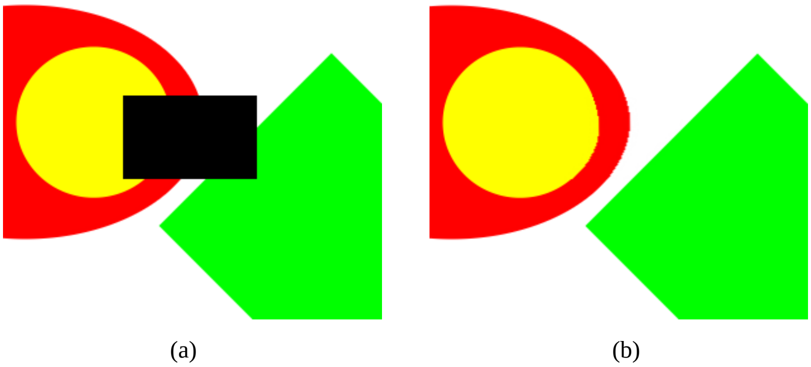


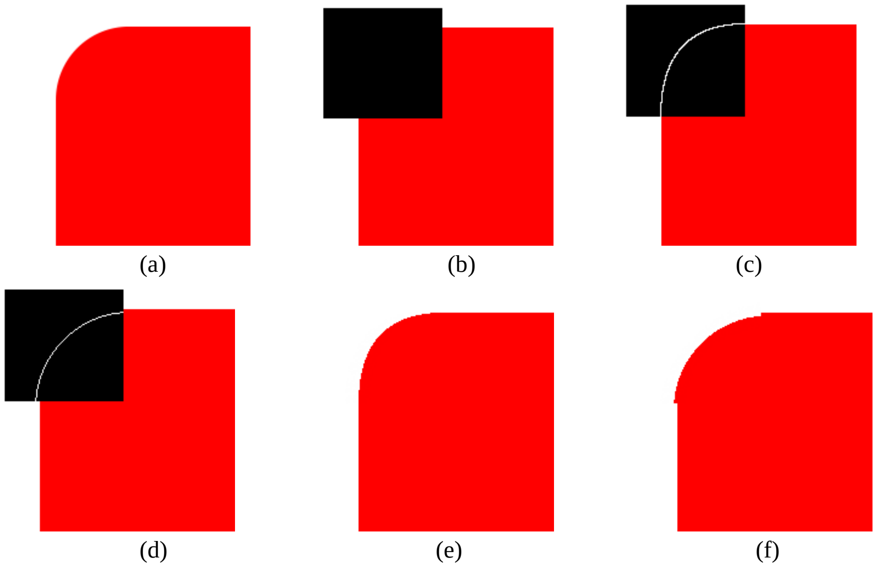
Fig. 11. Result of exemplar-based inpainting method. (a) Masked image. (b) Final result.

4. EXPERIMENTAL RESULTS

In this section, both of artificial and real images are used in the experiments to evaluate the effectiveness of the proposed Bezier-curve based hybrid inpainting method. The real images used in the experiments are obtained from Berkeley Segmentation Dataset [18].

4.1 Experiments on synthetic images

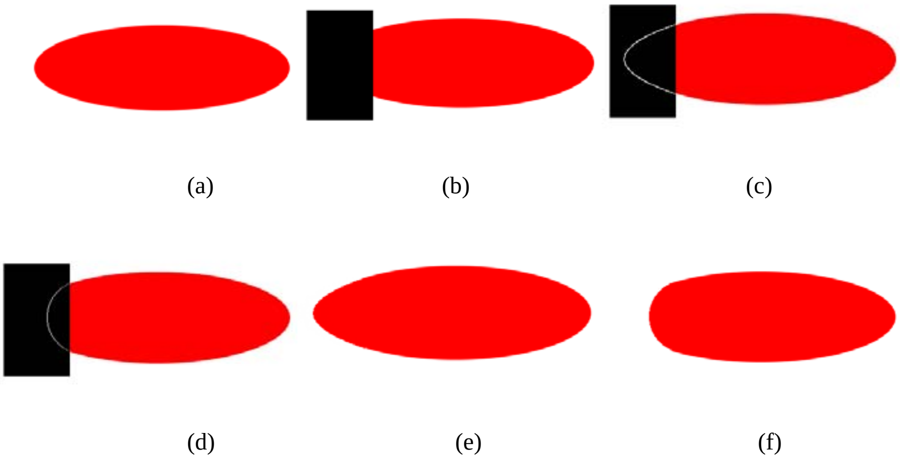
This work uses four experimental results on synthetic images to present the performance of our method. The first and second experiments demonstrate that using Bezier curves to connect the matched edges achieves better results than the one using the line and circle function to approximate the image structure [16]. In Fig. 12 and Fig. 13, we observe that the restored image skeletons affect significantly the final inpainting result. Figs. 12(e) and 13(e) show the inpainting results of the proposed method, and Figs. 12(f) and 13(f) are the results by the edge-based inpainting method [16]. It is obvious that the proposed method, in which the quadratic Bezier curves are used to reconstruct the edges in the missing area, outperforms the method in [16] which uses the circle function to restore the edges.



(a) (b) (c)

(d) (e) (f)

Fig. 12. The comparison of using Bezier Curves and locally circular shapes to restore the missing edges. (a) Original image. (b) The target region has been selected and marked with black rectangle. (c) The edge is restored with Bezier Curves. (d) The edge is restored with the function of circle. (e) The final inpainting result based on (c). (f) The final inpainting result based on (d).



(a) (b) (c)

(d) (e) (f)

Fig. 13. The comparison of using Bezier Curves and locally circular shapes to restore the missing edges. (a) Original image. (b) The target region has been selected and marked with black rectangle. (c) The edge is restored with Bezier Curves. (d) The edge is restored with the function of circle. (e) The final inpainting result based on (c). (f) The final inpainting result based on (d).

The image shown in Fig. 14(a) is a standard test image adopted in the exemplar-based image inpainting algorithm [14]. Fig. 14(b) is the masked image, and the black triangle is the missing area. Figs. 14(c) and 14(d) show the inpainting results by using the proposed algorithm and exemplar-based image inpainting method respectively. It demonstrates that our method restores correctly the image structure including straight lines and circles. The taper shape occurs when there is a curved structure included in the target region, it is because of the exemplar-based image inpainting method [14] is an isophote-driven method.

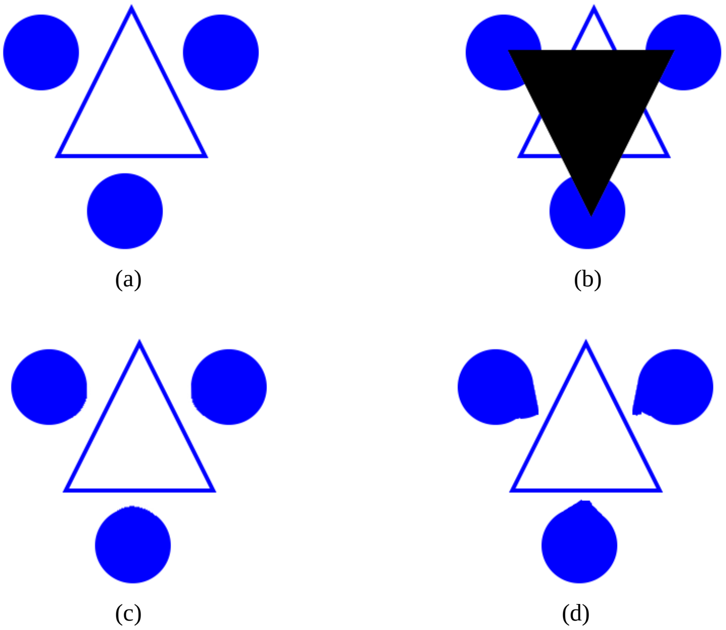
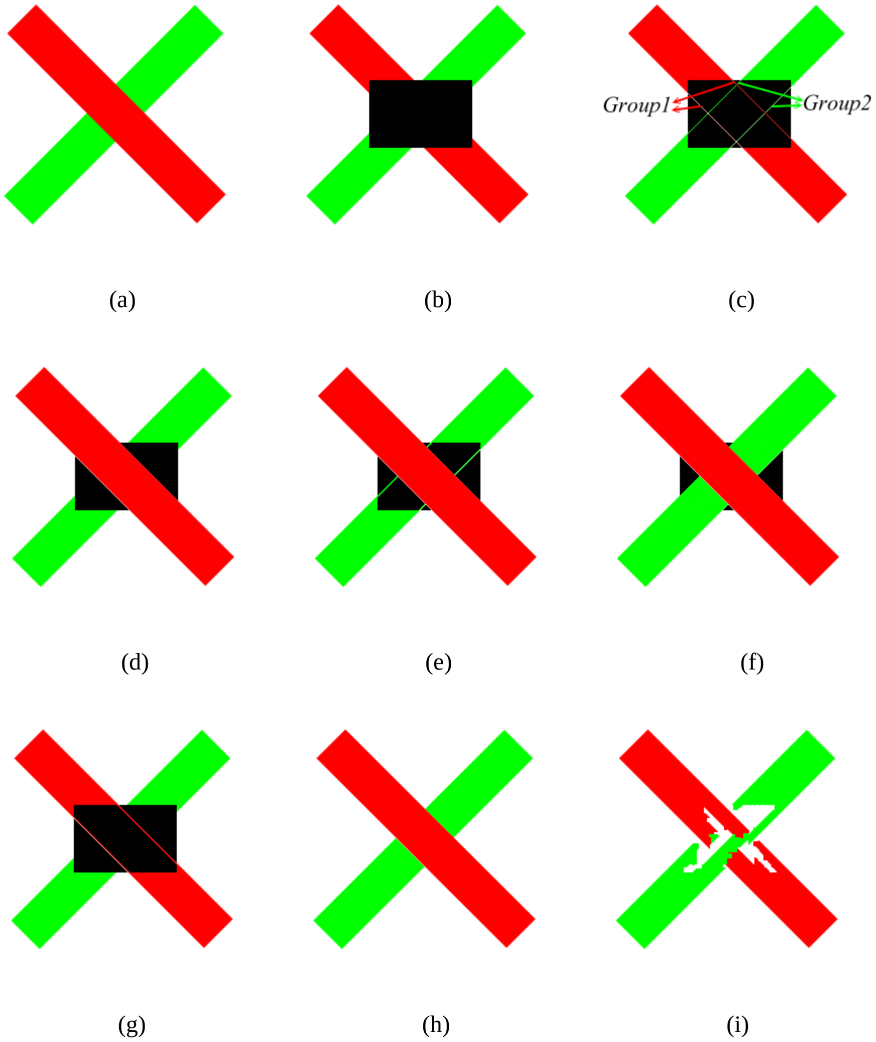


Fig. 14. The comparison of inpainting results between the proposed algorithm and the exemplar-based image inpainting [14]. (a) Original image. (b) Masked image. (c) The result of our method. (d) The result of exemplar-based image inpainting method.

Most inpainting approaches do not handle ambiguities in which the missing region covers the intersection of two regions as shown in Fig. 15(a). Our method considers the case of crossing couples. According to the edge grouping procedure, we find that the best configuration of Fig. 15(b) has two matched-edge groups, and every group includes two matched edges. When the number of matched-edge group is more than one, it means that the configuration in the missing area is cross, as shown in Fig. 15(c). In order to avoid producing broken result, we first repair the matched edges which belong to the first group and the first region which be surrounded by the edges of the first group. Fig. 15(d) is the reconstructed edges which belong to the first group, and Fig. 15 (e) show the inpainting result of the first region. After repairing the first region, we first reconstruct the edges in the second group as shown in Fig. 15(f), the second region which is surrounded by the edges of the second group is then restored as shown in Fig. 15(g). Finally, the remainder target region is filled using the exemplar-based image inpainting algorithm, the result is shown in Fig. 15(h). As for the inpainting order about the two groups, it can be decided arbitrarily. Since the exemplar-based image inpainting approach [14] does not consider the case of intersection, it produces a broken result as shown in Fig. 15(i). This comparison demonstrates the effectiveness of the proposed inpainting method.



(a) (b) (c)

(d) (e) (f)

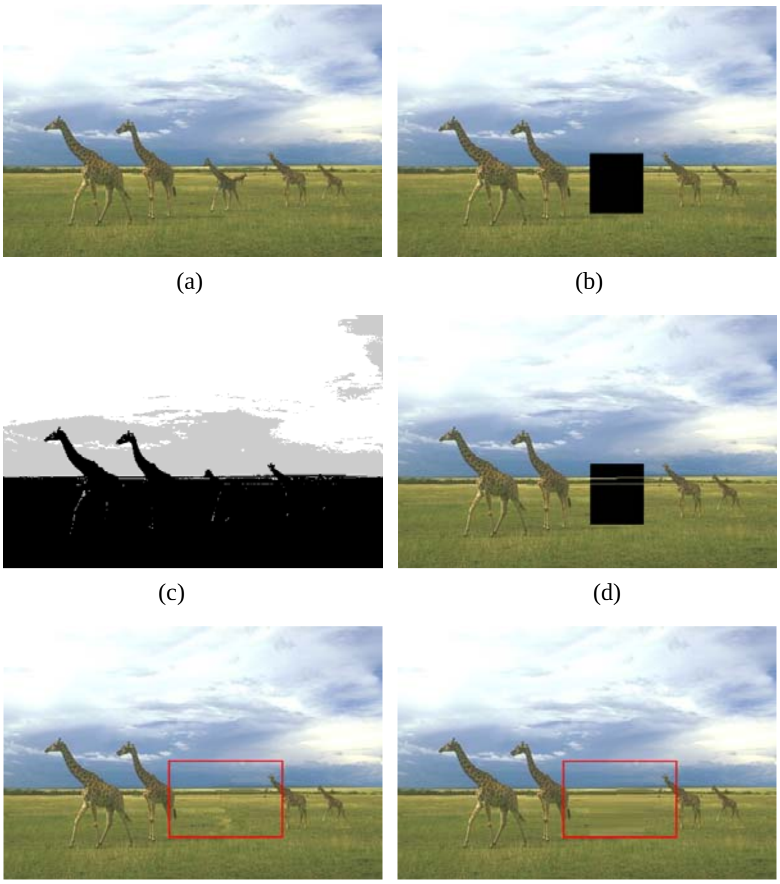
(g) (h) (i)

Fig. 15. Comparison between the proposed algorithm and exemplar-based image inpainting method. (a) Original image. (b) Masked image. (c)The result of the grouping procedure in edge matching step. (d) The result of the edges in the first group are restored. (e) The first region is restored. (f) The result of the edges in the second group are restored. (g) The intersection of two regions is restored. (h) The final result of our method. (i) The result of exemplar-based image inpainting.

4.2 Experiments on real images

The patch size of the exemplar-based inpainting method is decided by experience. If the missing area is highly textured, we should use the lager patch size when inpainting. The larger patch size is used, the more texture information is preserved. However, it takes long time in searching the best matched patch from the source region if the size of patch is large. In general, the patch size of 9×9 achieves a good tradeoff between in-painting performance and processing time for nature images.

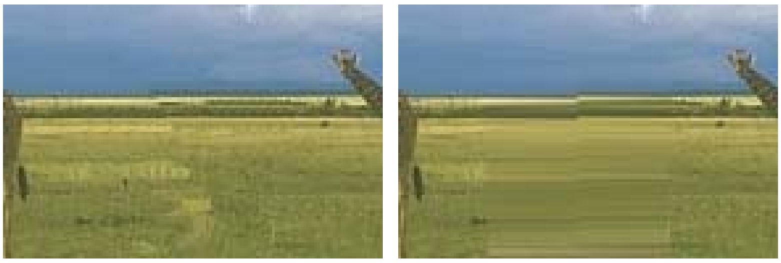
In this experiment, several real images are used to verify the performance of the proposed method. Figs. 16 and 17 show the comparison between the proposed algorithm and edge-based inpainting method [16]. It is obvious that the proposed method using the patch-based method to restore the images preserves texture information in the missing area. However, on the contrary, the textures of meadow and forest disappear in Figs 16(g) and 17(g) if only the pixel-based inpainting technique is used to restore the images. They demonstrate the effectiveness of the proposed hybrid inpainting method.



(a) (b)

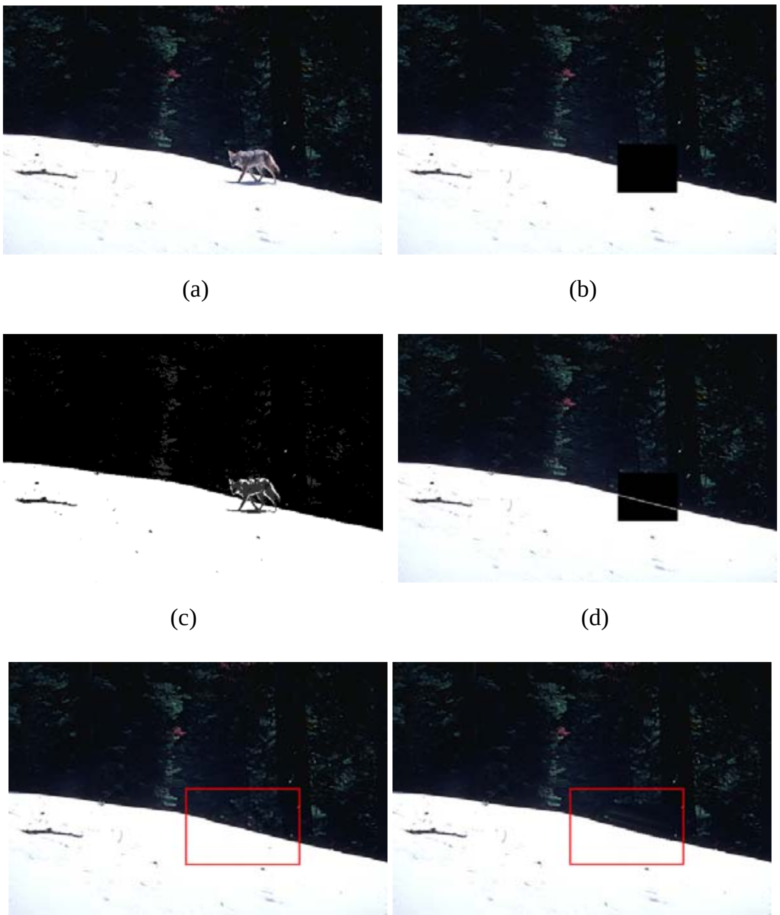
(c) (d)

(e) (f)



(g) (h)

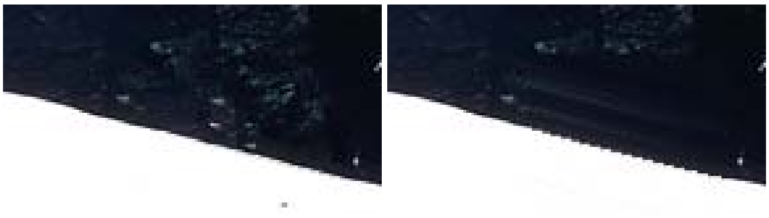
Fig. 16. Comparison between the proposed algorithm and edge-based inpainting method [16]. (a) Original image. (b) Masked image. (c) The segmentation result of the multilevel Otsu's thresholding (three-level). (d) The reconstructed edge image. (e) The result of our method. (f) The result of edge-based image restoration [16]. (g) Zoom-in on the areas of interest in (e). (h) Zoom-in on the areas of interest in (f).



(a) (b)

(c) (d)

(e) (f)

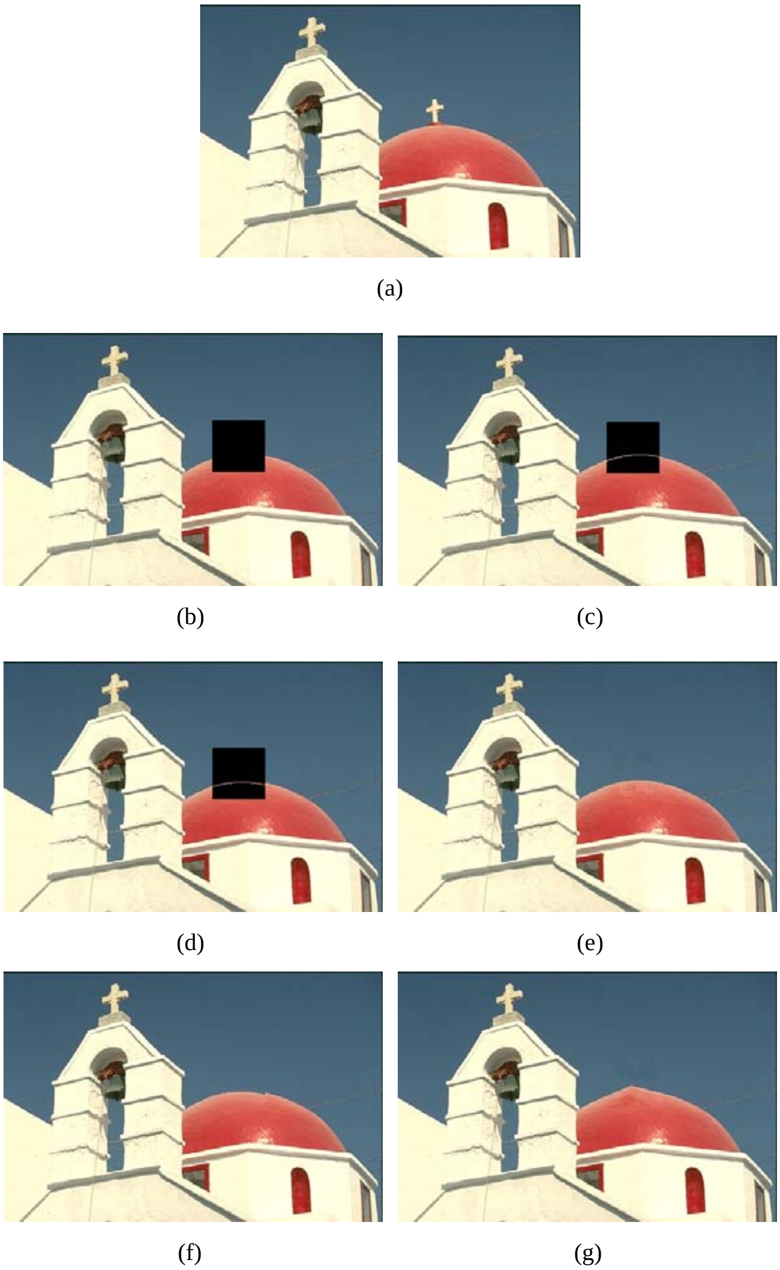


(g) (h)

Fig. 17. Comparison between the proposed algorithm and edge-based inpainting method [16]. (a) Original image. (b) Masked image. (c) The segmentation result of the multilevel Otsu's thresholding (three-level). (d) The reconstructed edge image. (e) The result of our method. (f) The result of using edge-based image restoration [16]. (g) Zoom-in on the areas of interest in (e). (h) Zoom-in on the areas of interest in (f).

In Fig. 18, our target is to remove the crucifix and reconstructs the curve structure of the red roof. Figs. 18(e), 18(f) and 18(g) illustrate the inpainting result by using the proposed method, edge-based inpainting method [16] and exemplar-based inpainting method [14] respectively. Compared Fig. 18(e) with Fig. 18(f), we find that the reconstructed edge by using the proposed Bezier-curve based method achieves better reconstruction result, it is due to the roof in the missing area cannot be approximated by a simple circle function. It is obvious that the taper effect occurs in Fig. 18(g), since the algorithm [14] cannot handle the curved structures.

Figure 19 shows a detail comparison between the proposed method and exemplar-based image inpainting approach [14] on "Lena" image. Figure 19(c) shows the segmentation result by the iterative image thresholding algorithm [17], which helps us obtain the edge information for restoration. The block line segments in Fig. 19(d) are the detected edges. The final inpainting result by the proposed method is shown in Fig. 19(g), and it demonstrates that the proposed method is able to reconstruct the curvature skeletons in the missing area.



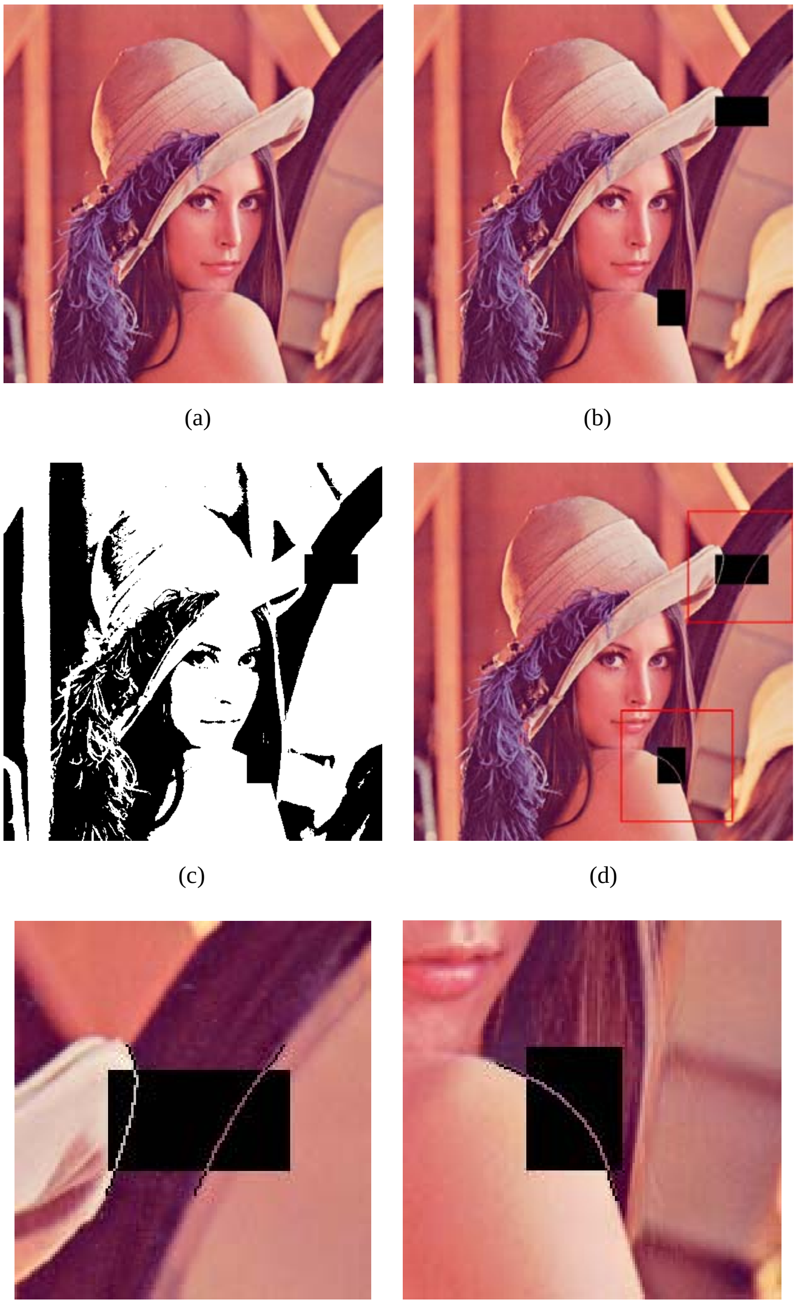
(a)

(b) (c)

(d) (e)

(f) (g)

Fig. 18. Comparison between the proposed method and other approaches in the literature. (a) Original image. (b) Masked image. (c) The edge is restored by Bezier curves. (d) The edge is restored by circles. (e) Inpainting result with the proposed method. (f) Inpainting result with edge-based inpainting method [16]. (g) Inpainting result with exemplar-based inpainting method [14].



(a)

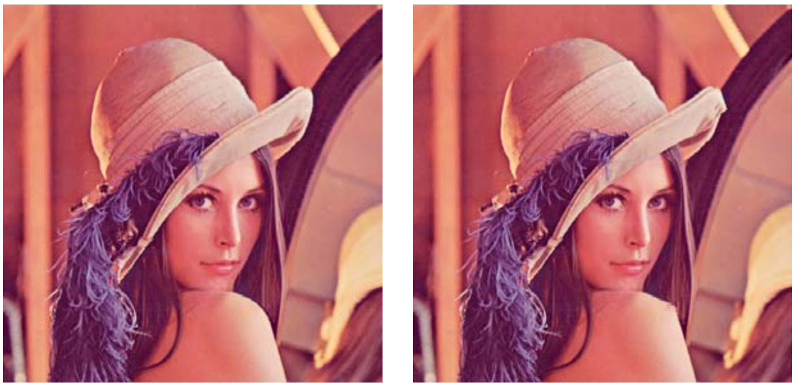
(b)

(c)

(d)

(e)

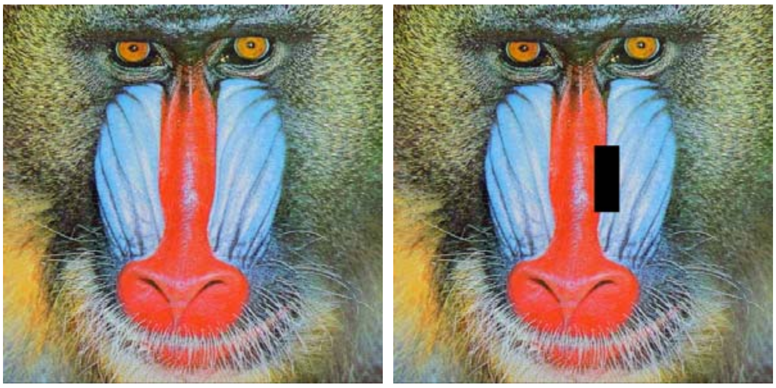
(f)



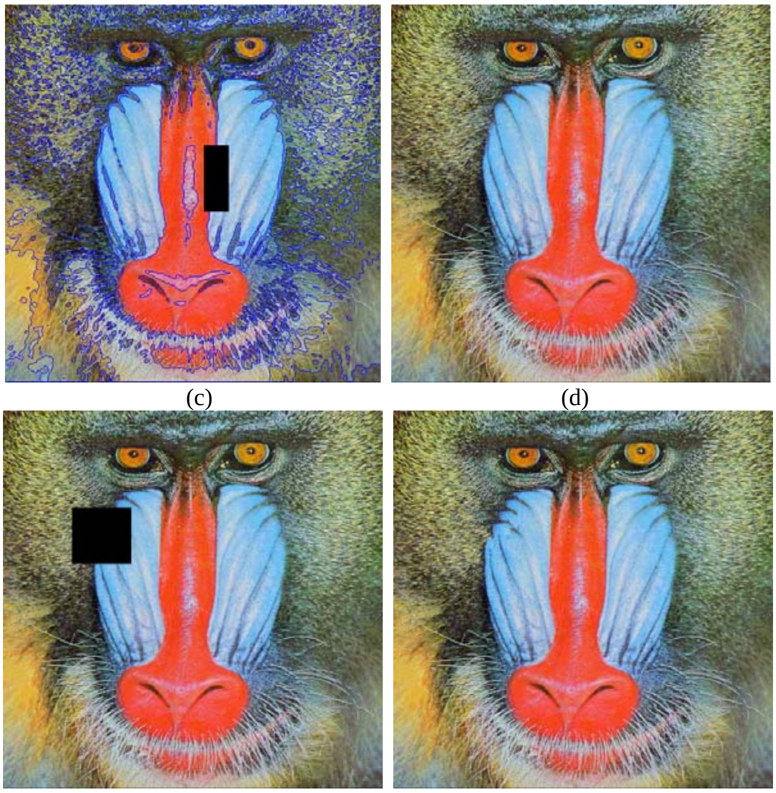
(g) (h)

Fig. 19. Comparison with exemplar-based image inpainting [14]. (a) Original image. (b) Masked image. (c) The segmentation result by the iterative image thresholding algorithm. (d) The reconstructed edge image by the proposed method. (e) Zoom-in on the areas of interest in (d). (f) Zoom-in on the areas of interest in (d). (g) The inpainting result with our method. (h) The result of exemplar-based image inpainting.

To evaluate the performance of the proposed method on the highly textured image, the image Baboon is used in the experiment. Fig. 20 shows the inpainting result of the proposed method. It is obvious that half of the mask in Fig. 20(e) is a highly textured area, the restored image by the proposed method is shown in Fig. 20(f), our method obtains the satisfactory result. The skeleton of the target region is build by quadratic Bezier curves, the textures in missing area is then restored by the patch-based inpainting method.



(a) (b)

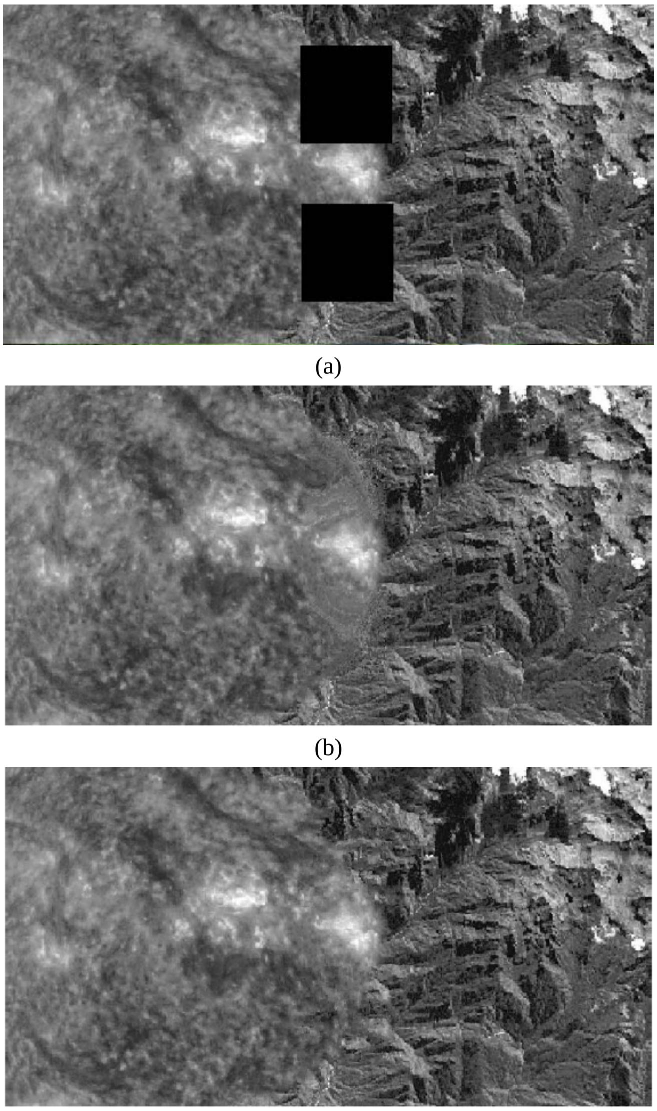


(c) (d)

(e) (f)

Fig. 20. The result of the proposed inpainting method on a highly textured image. (a) Original image. (b) Masked image. (c) The segmentation result of the multilevel Otsu's thresholding (three levels), in which the blue lines are the detected edges. (d) The result of our method. (e) Masked image. (f) The result of our method.

Figs. 21 and 22 show the comparison between the proposed method and the approach in [12] which reconstructed the image separately with structure and texture filling-in algorithms. Both methods perform well in the image with multiple textures as shown in Fig. 21. Since the edge matching and the skeleton reconstruct are included in the proposed method, the experimental result in Fig. 22 demonstrates that our proposed method obtains superior result in the image structure restoration of the missing area, as shown in Fig. 22(c). Finally Figs. 23 and 24 are the inpainting results of the proposed method on two real images. They show the effectiveness of the proposed method.



(a)

(b)

(c)

Fig. 21. Comparison with the approach in [12]. (a) Masked image. (b) The result of the approach in [12]. (c)The inpainting result of our method.

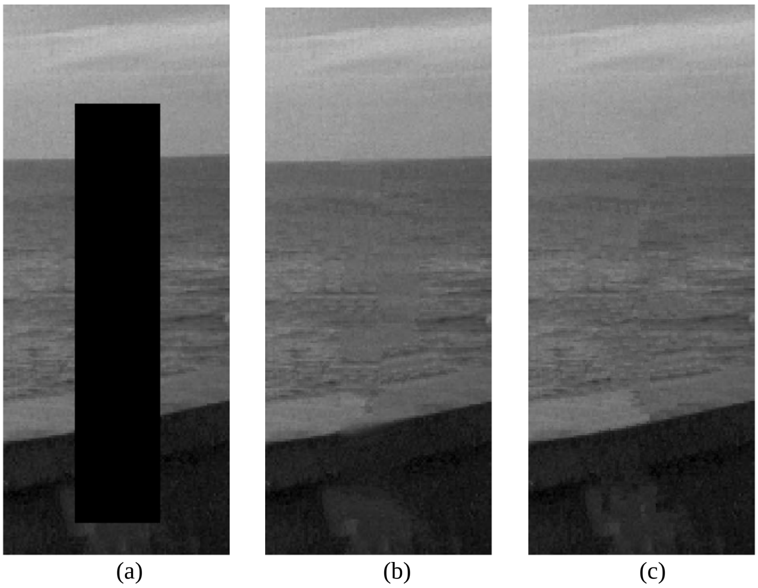


Fig. 22. Comparison with the approach in [12]. (a) Masked image. (b) The result of the approach in [12]. (c)The inpainting result of our method.

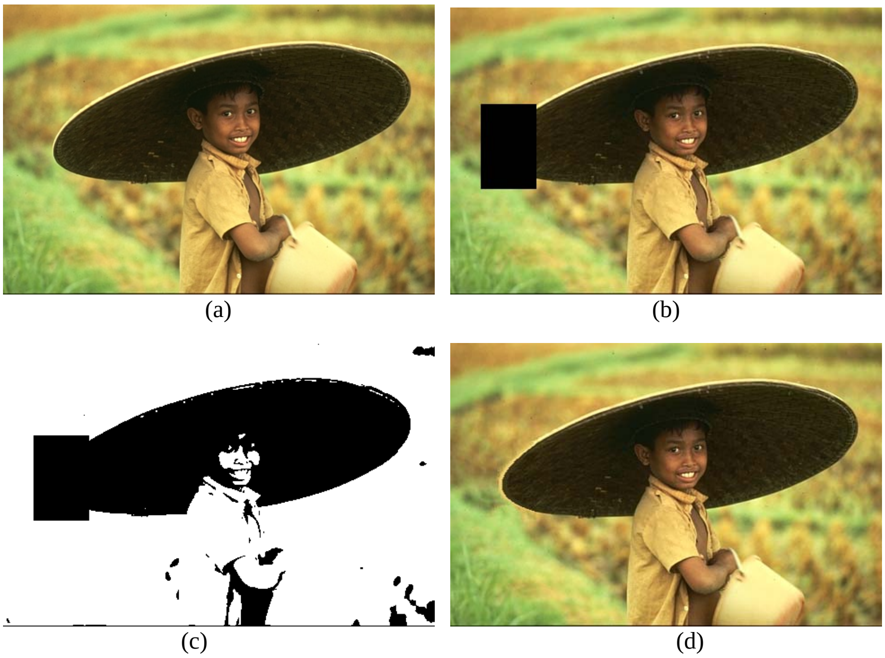
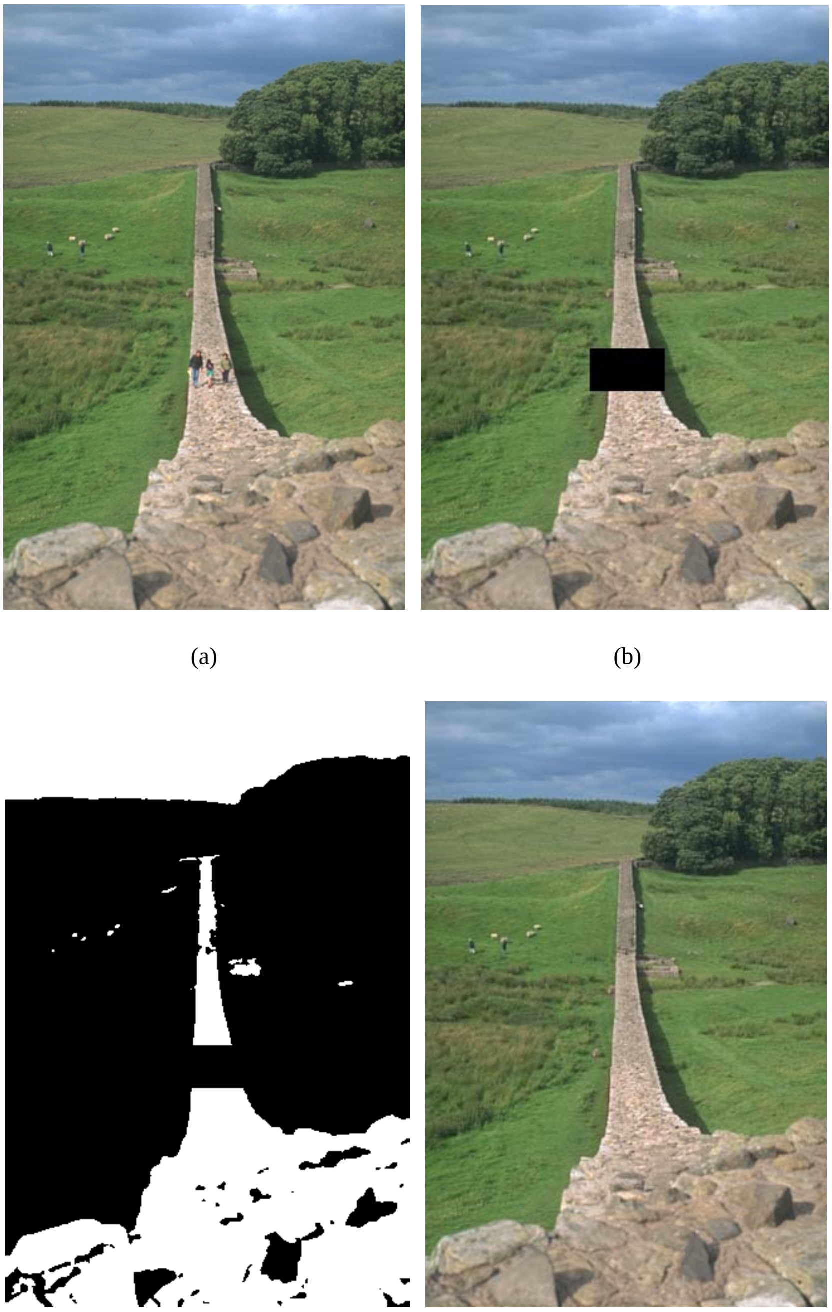


Fig. 23. The result of the proposed inpainting method. (a) Original image. (b) Masked image. (c) The segmentation result by the iterative image thresholding algorithm. (d) The result of our method.



(a) (b)

(c) (d)

Fig. 24. The result of the proposed inpainting method. (a) Original image. (b) Masked image. (c) The segmentation result by the iterative image thresholding algorithm. (d) The inpainting result of our method.

All experiments in this work are preformed on a machine with AMD phenom II X4 965 3.4GHz CPU and 4G RAM. Table 1 shows the comparison of the processing time between the proposed method and the exemplar-based approach[14], it is observed that our approach takes longer processing time than the exemplar-based method. The reason is that the edge matching procedure is used in our method for restoring the skeleton of the target region, and it is time-consuming. Compared to the edge-based inpainting method[16], our method uses the patch-based inpainting method for preserving texture information, while the pixel-based filling method is utilized in the edge-based method. The patch-based inpainting method needs to search the source region for finding the most similar texture by calculating the sum of squared differences (SDD) between the source and target patches, it is time-consuming. Therefore our method is also slower than the edge-based image restoration algorithm.

Table 1. The comparison of processing time between the proposed method and the exemplar-based method. (seconds)

Image	Fig.16(b)	Fig. 17(b)	Fig. 18(b)	Fig. 19(b)	Fig. 23(b)	Fig. 24(b)
Exemplar-based	2.64	2.72	1.68	1.82	2.79	1.90
Proposed	3.07	3.02	2.05	2.14	3.00	2.23

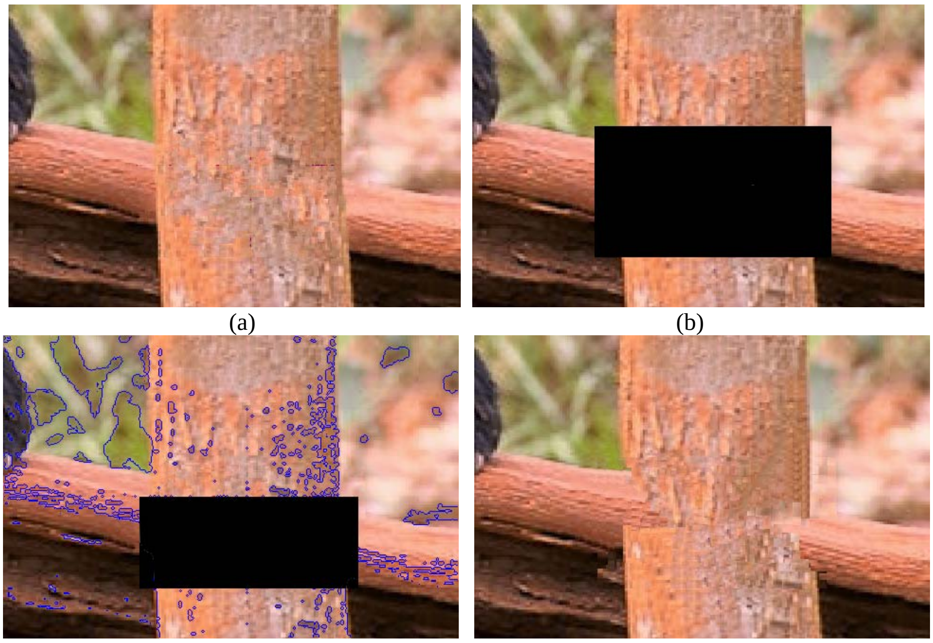

(a)
(b)
(c)
(d)

Fig. 25. The result of the proposed inpainting method. (a) Original image. (b) Masked image. (c) The segmentation result of the multilevel Otsu’s thresholding (three levels), in which the blue lines are the detected edges. (d) The inpainting result of our method.

Our approach also shares the common limitation with the edge-based inpainting approach, it has no ability to deal with the problem of depth ambiguities where the missing area covers the intersection of two perpendicular regions as shown in Fig. 25(a), especially when the color of two cross objects is similar. This limitation can be partially solved by interactive user inputs [15].

5. CONCLUSIONS

Image inpainting technique has wide applications range from restoration of photographs, films and paintings, to removal of occlusions, such as text, subtitles, stamps and publicity from images. This paper presents a new image inpainting method based on Bezier curves and the edge-based image restoration algorithm. The steps of edge detection, skeleton restoration and image inpainting are improved. The main contributions of the proposed method include: (1) Iterative Otsu's thresholding method is utilized for segmentation and the edge information is then obtained, this method is efficient and overcomes the oversegmentation problem; (2) A Bezier curves based reconstruction method is proposed to proposed to connect the matched edges in the missing areas, which preserves successfully the curvature structures in the damaged image and achieves better results than the conventional edge-based restoration method; and (3) A hybrid approach is adopted in the inpainting process. The skeletons in the missing areas are first depicted by using the pixel-based inpainting approach. In order to preserve the texture information of image, a patch-based inpainting method is then used to fill the missing objects. Experimental results on both synthetic and real images have shown that the proposed algorithm outperforms other existing inpainting algorithms in recovering missing objects in images.

REFERENCES

1. M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proceeding ACM Conference Computer Graphics*, 2000, pp. 417–424.
2. T. F. Chan and J. Shen, "Mathematical Models for Local Deterministic Inpaintings", *Technical Report CAM 00-11, Image Processing Research Group, UCLA*, 2000.
3. T. F. Chan and J. Shen, "Non-Texture Inpainting by Curvature Driven Diffusions (CDD)", *Journal of Visual Communication and Image Representation*, 2001, pp.436-449.
4. A. Efros and T. Leung, "Texture synthesis by nonparametric sampling," in *Proceeding International Conference Computer Vision*, 1999, pp. 1033-1038.
5. L. Y. Wei and M. Levoy, "Fast texture synthesis using tree-structured vector quantization," in *Proceeding ACM Conference Computer Graphics*, 2000, pp. 479–488.
6. M. Ashikhmin, "Synthesizing natural textures," in *Proceeding ACM Symposium Interactive 3D Graphics*, 2001, pp. 217–226.
7. S. C. Pei, Y. C. Zeng, and C. H. Chang, "Virtual restoration of ancient Chinese paintings using color contrast enhancement and Lacuna texture synthesis," *IEEE*

Transactions on Image Processing, Vol. 13, 2004, pp. 416–429.

8. A. Efros, and W.F. Freeman, "Image quilting for texture synthesis and transfer," in *Proceeding ACM Conference Computer Graphics*, 2001, pp. 341-346.
9. L. Liang, C. Liu, Y.Q. Xu, B. Guo, and H.Y. Shum, "Real-time texture synthesis by patch-based sampling", *ACM Transactions on Graphics*, Vol. 20, 2001, pp.127-150.
10. V. Kwatra, A. Schodl, I. Essa, G. Turk, and A. Bobick, "Graphcut textures: Image and video synthesis using graph cuts," *ACM Transactions on Graphics*, 2003, pp. 277–286.
11. Q. Wu and Y. Yu, "Feature matching and deformation for texture synthesis," *ACM Transactions on Graphics*, Vol. 23, 2004, pp. 364–367.
12. M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, "Simultaneous structure and texture image inpainting," *IEEE Transactions on Image Processing*, Vol.12, 2003, pp. 882–889.
13. I. Drori, D. Cohen-Or, and H. Yeshurun, "Fragment based image completion", *ACM Transactions on Graphics*, Vol. 22, 2003, pp. 303–312.
14. A. Criminisi, P. P'erez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Transactions on Image Processing*, Vol. 13, 2004, pp. 1200–1212.
15. J. Sun, L. Yuan, J. Jia, and H. Y. Shum, "Image completion with structure propagation," *ACM Transactions on Graphics*, 2005, pp. 861–868.
16. A. Rares, M.J.T. Reinders, and J. Biemond, "Edge-based image restoration," *IEEE Transactions on Image Processing*, Vol. 14, 2005, pp. 1454-1468.
17. L. Dong, G. Yu, P. Ogunbona and W. Li, "An efficient iterative algorithm for image thresholding," *Pattern Recognition Letters*, Vol. 29, 2008, pp.1311-1316.
18. Test images are from Berkeley Segmentation Dataset: Images available at http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/
19. J. C. Hung, C. H. Hwang, Y. C. Liao, N. C. Tang and T. J. Chen, "Exemplar-based image inpainting based on structure construction," *Journal of Software*, Vol. 3, No. 8, 2008, pp. 57-64.

20 A. Wong and J. Orchard, "A nonlocal-means approach to exemplar-based inpainting," *Proceeding of IEEE International Conference on Image Processing*, 2008, pp. 2600-2603.

21. Z. Xu, and J. Sun, "Image inpainting by patch propagation using patch sparsity," *IEEE Transactions on Image Processing*, Vol. 19, No. 5, 2010, pp. 1153-1165.
22. J. Shen, X. Jina, C. Zhou, C. L. Wang, "Gradient based image completion by solving the Poisson equation," *Computers & Graphics*, Vol. 31, 2007, pp. 119-126.
23. N. Komodakis and G.. Tziritas, "Image completion using efficient belief propagation via priority scheduling and dynamic pruning," *IEEE Transactions on Image Processing*, Vol. 16, No. 11,2007, pp. 2649-2261.