A Region Boundary Reconstruction Method for Structure-Aware Image Inpainting

Takayuki KIMURA*, Kazu MISHIBA*, Shunsuke IWAMURA†, Keiichiro SHIRAI‡, and Masaaki IKEHARA*
* Faculty of Engineering, Keio University, Yokohama, Kanagawa, 223-8522 Japan
E-mail:{kimura, mishiba, ikehara}@tkhm.elec.keio.ac.jp
† Broadcast Engineering Department, NHK, Japan
‡ Faculty of Engineering, Shinshu University, Nagano, Japan

***Abstract*—This paper presents a structure-aware inpainting, which automatically reconnects broken links of boundaries on a missing region, then inpaints each remaining part by exemplars.This paper presents a structure-aware inpainting, which automatically reconnects broken links of boundaries on a missing region, then inpaints each remaining part by exemplars. Boundary contours obtained by a region segmentation are connected by Bézier curves. We show mainly how to restore curved structures on the target region. To restore the accurate curved structures, the pair of contours and control points of a Bézier curve are decided. The control points enable to reconstruct the quadratic and cubic curved structures. Reconstructed structures help to restore target region and a natural result is obtained.**

I. Introduction

Recently, the image inpainting method which restores missing regions of an image becomes widely known and is implemented on major software for the purpose of removing unnecessary objects. The algorithms are classified into two types, the diffusion-based methods and the exemplar-based methods. Diffusion-based methods are often used when a target region is narrow. Exemplar-based methods are applied to large or textured restoration targets. In this paper, we consider the restoration of large hole region and discuss the latter exemplar-based methods.

The exemplar-based methods [1], [2], [3] fill in holes by pasting similar patch images from surrounding regions, and are suited for the restoration of large holes. Because pasting patches might have textured patterns if the surrounding regions include textured regions. The still remaining problem of the exemplar-based methods is restoring missing boundary. The existing methods are unable to reconnect complex structures like curved edges.

To restore the various missing structures (i.e. outline form of an object), [4] specifies the structures manually and restores them. In this method, tentative lines are drawn manually, and the missing structures are restored tracing the lines by exemplar-based method. Though this method can be applied to a number of structure constructions, the structures have to be specified by hand.

The automatic curved structure construction and inpainting is proposed by Hung [5]. This method extracts missing structures and restores them by using Bézier curves. The remaining target region is restored by the exemplar-based inpaining along the recovered structures. However, this method cannot restore the curved structures in some cases.

If a curved structure is interrupted by a damaged region, there must be two broken link boundaries around the region. The broken link boundaries are called "contour edges" and each contour edge has to be reconnected to reconstruct curved structures. In order to reconnect the contour edges exactly, the accurate pairs of contour edges have to be determined. While the algorithm used in [5] determine the correct pairs in case one contour edge is obviously narrowed down as the candidate of connecting another contour edge by spatial location. Meanwhile, if candidates of a connecting contour edge are not narrowed down to one candidate by spatial location, the correct pairs are not determined.

In this paper, we present a novel inpainting method to restore the curved edges naturally and automatically. Our structure construction is based on [5] method, and uses the position of contour edges and similarity of colors to decide correct pairs of the contour edges. The similarity of colors is calculated on color segmented image. These two factors enable to determine more accurate pairs of contour edges. In addition, we define the five control points which are needed to draw the cubic line with Bézier curve algorithm to draw the cubic lines. By these two improvements, the proposed method can acquire accurate restored curved structures and natural completed image.

The rest of this paper is organized as follows. Sec. II shows related works, and Sec. III describes the proposed method. Sec. IV shows the experimental results and comparisons.

II. Related works

How to fill in the target region which has curved structures is proposed in [5]. This method achieves restoration of the damaged region including missing curves by automatically recovering the curved structures as preprocessing. First, the edge detection is performed by a mean shift segmentation. After detecting edges, lines are extended from end of each contour edges to target region Fig.1(a). Then intersection points of each line are found and set as candidates of pairs. The determined pairs of contour edges are reconnected by Bézier curve algorithm. Finally, their exemplar-based inpainting is applied to remaining Ω and whole regions are completed.

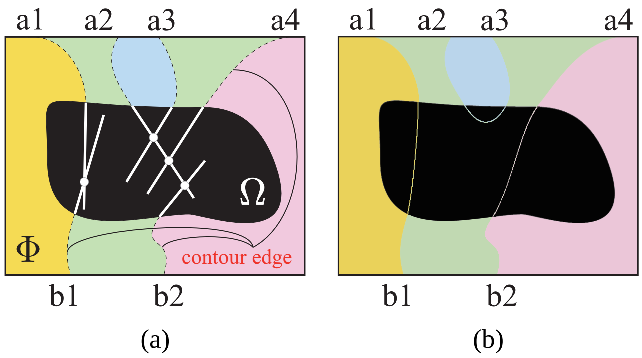


Fig. 1. Concept of curved structure restoring: The dashed lines trace the boundary contours (a), and white lines are extended lines from each end of contour, and white points are intersection points, (b) is our result of the curved structure restoring.

Fig. 2. Color Segmentation results. (a) Original image. (b) Segmented regions. The black region is target region.

Above mentioned, it is difficult to determine the correct pairs of contour edges in some cases by [5] method. From an objective viewpoint, it seems that the lines a1-b1, a2-a3, a4-b2 should be connected like Fig.1(b). However, boundaries a2-a3 and a4-b2 are not restored because the pairs of connecting contour edges cannot be determined. This method determines the contour pairs by existence of intersection points. The contour pairs are determined by the existence of single intersection point, therefore this method cannot deal with multiple intersection points (Fig.1(a)). In order to restore various curved structures, we need other conditions.

III. PROPOSED ALGORITHM

In this section, we present a novel inpainting method based on restored curved edges. The proposed method achieves the various curved structure constructions and generates the fine completed image. Our method consists of following steps.

(1) Color segmentation
(2) Contour connection
(3) Exemplar-based inpainting

(1) In color segmentation, an input image is separated by color and transformed into color segmented image (III-A).

(2) Contour connection has three steps. First, to find the intersection points, lines are extended from end of each contour edge along the calculated slope (III-B). Second, pairs of contour edges are determined by intersection points and similarity of colors (III-C). Finally, curved structures are restored by two different methods (III-D and III-E).

(3) Remaining target regions are completed along the restored structures by our exemplar-based methods (III-F).

A. Region Segmentation to obtain boundary contours

To separate regions by color, we apply a color segmentation method described in [6]. First an image is smoothed by using a variant of anisotropic diffusion [7]. After smoothing, each pixel is assigned its own segment, and the initial segments are connected with its 4 neighboring segments. If the average color of two neighboring segments is similar, a small size segment is merged to a large size segment iteratively. Small segments less than 100 pixels are merged to the neighboring large segments. A result of this segmentation algorithm is shown in Fig.2. Finally, pixels on the boundary of segments are used as elements of contour edges.

B. contour edge extension for the crossing test

The slope direction of a contour edge (Fig.3(a)) is defined by using coordinates of points $\{\mathbf{c}_{i,n}\}_{n=1}^{kN}$ on the i'th contour edge as follows:

$$\mathbf{s}_i = \frac{1}{K}\sum_{n=0}^{N-1}\frac{1}{2^{n-1}}(\mathbf{c}_{i,kn} - \mathbf{c}_{i,k(n+1)}), \tag{1}$$

where $\mathbf{c}_{i,0}$ is coordinate of end point of the i'th contour edge and $\mathbf{c}_{i,n+1}$ is neighboring coordinate of $\mathbf{c}_{i,n}$ on the i'th contour edge, and k controls the sampling interval and we set $k = 4$ to reduce the noise effects. Additionally, because a $\mathbf{c}_{i,n}$ is far from the end point $\mathbf{c}_{i,0}$ is unreliable, the weight $1/2^{n-1}$ decreases with the increasing distance from the end point. The number of samples is $N = 4$ in our experiment. $K = \sum_{n=0}^{N-1}\frac{1}{2^{n-1}}$ is a normalization factor. Using the obtained slope $\mathbf{s}_i$ and extending lines from $\mathbf{c}_{i,0}$, an intersection of two lines is determined.

C. Determining pairs of contour edges

Within some intersections of extended contour edges, correct pairs need to be found for the better boundary reconstruction. To find pairs, we use two assumptions. (1) Extended lines from the same object have similar colors (a set of two colors on both sides of a boundary), (2) the position of an intersection is near by end points.

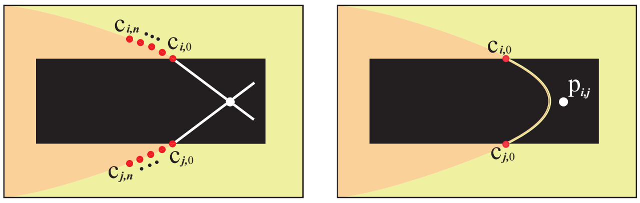
First, we consider all possible combinations of pairs, and then calculate the similarity of colors as follows:

$$P_c = e^{\frac{1}{2\sigma_c^2}\{\|I(\mathbf{c}_{i,0})_A - I(\mathbf{c}_{j,0})_A\|^2 + \|I(\mathbf{c}_{i,0})_B - I(\mathbf{c}_{j,0})_B\|^2\}}. \tag{2}$$

where $I(\mathbf{c}_{\{i,j\},0})$ shows the color of end points of contours i, j and A, B show the corresponding side of a boundary. Then, we calculate the distance among an intersection point $\mathbf{p}_{i,j}$ and two end points:

$$P_d = e^{\frac{1}{2\sigma_d^2}\{\|\mathbf{p}_{i,j} - \mathbf{c}_{i,0}\|^2 + \|\mathbf{p}_{i,j} - \mathbf{c}_{j,0}\|^2\}}. \tag{3}$$

Using above two criteria and multiplying them P_c, P_d, we sort the order of combinations in descending order, and pick up a good pair from upper to lower values iteratively.



(a) (b)

Fig. 3. The quadratic curve structure construction with three control points: (a) Crossing test of two extended lines estimated by sampling points on contour curves. (b) Drawn quadratic Bézier curve by three control points: an intersection and two end points of contours.

D. Quadratic curve structure for crossing convex boundaries

For the contour pairs which have an intersection, we use conventional method [5] to draw boundary edges. An obtained intersection $\mathbf{p}_{i,j}$ and two end points $\mathbf{c}_{i,0}, \mathbf{c}_{j,0}$ are used as control points of Bézier curve:

$$B(t) = (1-t)^2\mathbf{c}_{i,0} + 2t(1-t)\mathbf{p}_{i,j} + t^2\mathbf{c}_{j,0}, \;\; t \in [0,1]. \quad (4)$$

As a result a simple quadratic curve is obtained as shown in Fig.3(b).

E. Cubic curve structure for non-crossing concave boundaries

When the extended contour edges have no intersections but have a combination of similar colors, there is a possibility the boundary should be connected but the shape is convex and little squiggly. In this case, the following additional control points help to construct a desired boundary by Bézier curve as shown in Fig.4.

Two extended contour edges are culled with arbitrary length (we set 1/4 of distance between two end points), and the opposite end points of the line segments $\mathbf{p}_{i,m}, \mathbf{p}_{j,m}$ are newly taken as control points. Then a final center control point is given as the mean position of above two points: $\mathbf{p}_{i,j} = \frac{1}{2}(\mathbf{p}_{i,m}+\mathbf{p}_{j,m})$. The cubic Bézier curve using five control points is given as follows:

$$\begin{array}{l} B(t) = (1-t)^4\mathbf{p}_{i,0} + 4(1-t)^3 t\mathbf{p}_{i,m} \\ \quad +6(1-t)^2t^2\mathbf{p}_{i,j} + 4(1-t)t^3\mathbf{p}_{j,m} + t^4\mathbf{p}_{j,0} \end{array}, \;\; t \in [0,1] \quad (5)$$

As a result a cubic curve is obtained as shown in Fig.4(d).

F. *Exemplar-based inpainting* [2] along restored contours

In previous sections, boundaries of regions are reconstructed. Using these reconstructed boundary information, we finally fill in the remaining hole regions by using an exemplar-based inpainting.

In our method, each region is separated by the boundary of color segments, and regions on a hole tend not to have a relation each other, that is neighboring regions have different textures in many cases as shown in Fig.5(a), therefore we inpaint each hole Ω_i separately by using exemplars from the same segmented region R_i. As the inpainting method, we use Criminisi's [2] and modify it so as to decide the inpaint order of pixels and look for the patch similarity only by using the information of the same segment. For example, when we

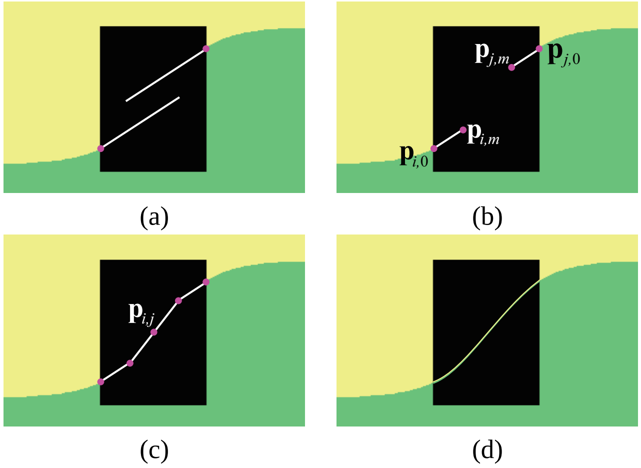
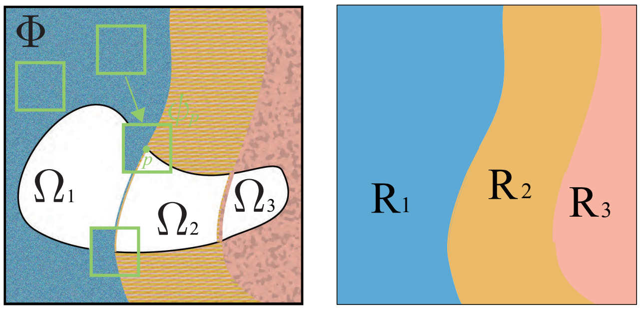


Fig. 4. The cubic curve structure construction with three control points: (a) Crossing test of two extended lines along contour curves. (b) Culling of extended lines, and four additional end points on line segments as control points. (c) A mid point as a final control point. (d) Drawn cubic Bézier curve by five control points.



(a) (b)

Fig. 5. Our exemplar-based inpainting: Each hole region (a) is separately inpainted by using exemplars from the same segmented region (b). p is a target point and ψ_p is a target patch (a). Ω_1, Ω_2 and Ω_3 are divided target regions by segmented region and restored structures (a). R_1, R_2 and R_3 are segmented region and each index links the indexes of Ω_i (b).

restore Ω_1, pasting patch is found on coordinates on R_1 region and the patch is filled in only on Ω_1 in a target patch ψ_p (Fig.5).

IV. Experimental Results

We apply our method to some images which include curved structures and verify the results. In our experiment, the size of patch is set to 9×9 pixels and the color is RGB normalized within $[0, 1]$.

Fig.6 show our results and the process of our experiment. (a) shows a damaged image, and restoration of curved structures is represented in (b). The four contour edges around damaged regions could be restored without mistake to connect. These curved structures are not reconstructed by [5] because the extended lines from each contour edges have a number of intersection points. The result of our exemplar-based method is obviously restored separately (c). (d) shows a result of [1]. The undesirable result is acquired since the curved structures are not restored as preprocessing.

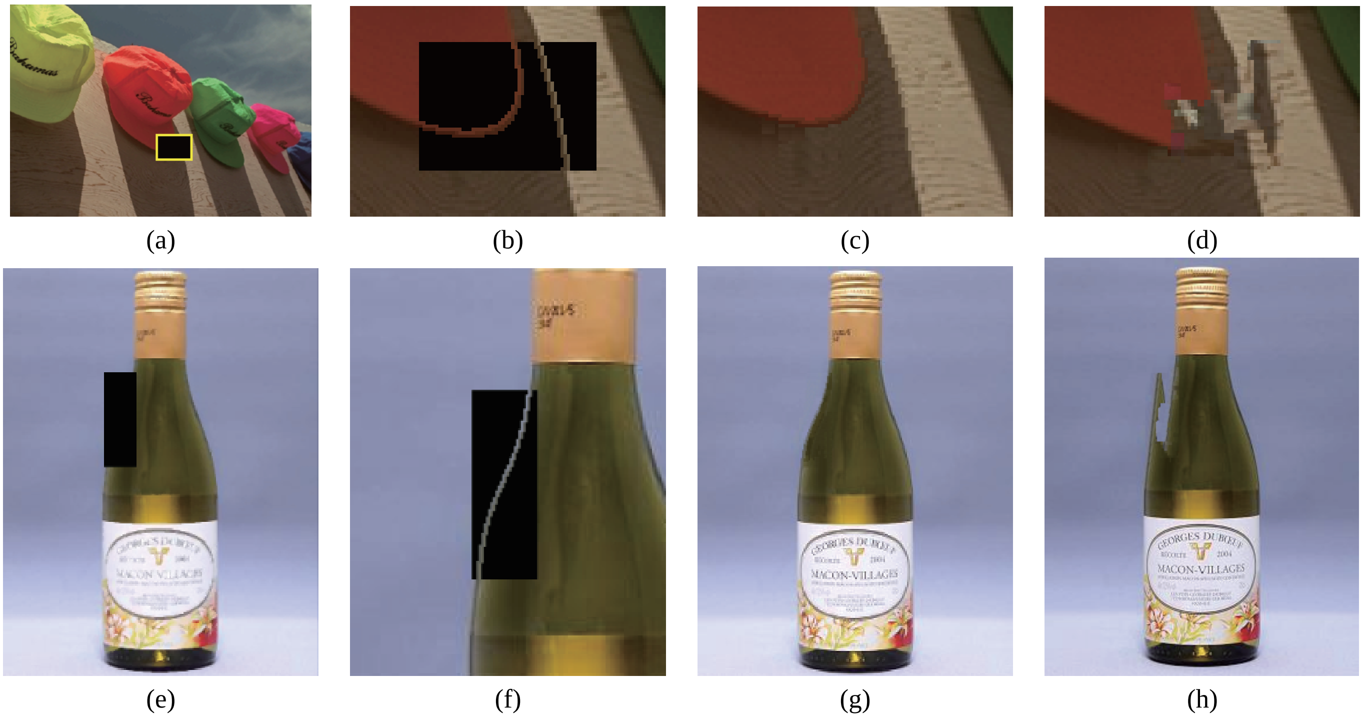


Fig. 6. Result of curved structure construction and restoring target region along the recovered curved structures: (a) and (e) are damaged image and the black regions are damaged regions. (b) and (f) The curved structures are naturally described and combinations of lines are correct. (c) and (g) The result of our algorithm. (d) and (h) The result of [1] and the curved structures are not restored as preprocessing.

Fig.6(e)-(h) shows a restoration of cubic curve and recovered remaining region. (e) shows a damaged image, and restoration of cubic curve is represented in Fig.6(f). (g) is the result of our inpainting method. (h) shows a result of [1]. As shown in (f), the cubic curve is restored smoothly, and the result can be restored naturally (g).

Above results show our structure construction effectiveness. The pairs of contour edges are determined correctly, and the quadratic and cubic formed curves are drawn naturally. The results of our exemplar-based method are restored properly from an objective viewpoint. The restored curves on the enlarged image are seemed jagged curve in our algorithm. Improving this problem is the challenging task.

V. Conclusion

We have presented a structure-aware inpainting method, especially showed an automatic pair selection method of broken boundaries, and showed the boundary interpolation method adapting to natural concave curves. By using this method, one can restore objects which conventional methods cannot deal with more naturally.

The still remaining problem of our method is the accuracy of segmentation. Since proposed method use simple color segmentation, if there are colorful pixels around contour edges, regions are not separated into recognizable segments. In such a case, the boundary pair selection method might fail to find correct pairs. Using texture segmentation is one of the future works to improve the accuracy of inpainting.

References

[1] A. Criminisi, P. Perez, and K. Toyama, "Object removal by exemplar-based inpainting," in *Proc. of IEEE Computer Vision and Pattern Recognition*, 2003, pp. 721–728.

[2] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans on Image Processing*, vol. 13, no. 9, pp. 1200–1212, 2004.

[3] I. Drori, D. Cohen-Or, and H. Yeshurun, "Fragment-based image completion," *In ACM Trans. on Graphics (SIGGRAPH 2003 issue)*, vol. 22, pp. 303–312, 2003.

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

[5] J. C. Hung, C.-H. Hwang, Y.-C. Liao, N. C. Tang, and T.-J. Chen, "Exemplar-based image inpainting base on structure construction," *Journal of Software*, vol. 3, no. 8, pp. 57–64, 2008.

[6] C. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," in *ACM Trans. on Graphics (SIGGRAPH)*, 2004, vol. 23, pp. 600–608.

[7] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, 1990.