Fast and Structure-Preserving Inpainting Based on Probabilistic Structure Estimation

Takashi Shibata Akihiko Iketani Shuji Senda
NEC Corporation, 1753 Shimonumabe, Nakahara-ku, Kawasaki, Kanagawa, 211-8666, Japan
t-shibata@hw.jp.nec.com iketani@cp.jp.nec.com s-senda@ap.jp.nec.com

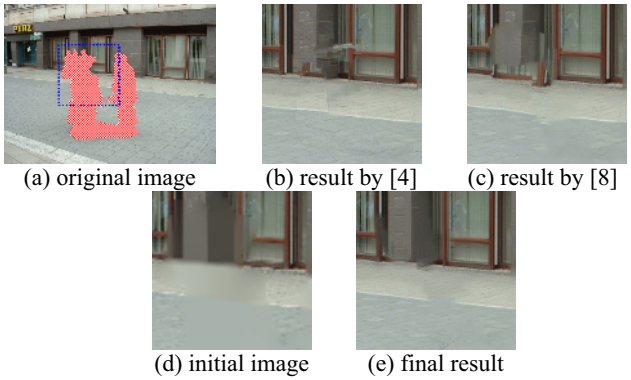
Abstract

This paper presents a novel inpainting method based on structure estimation. The method first estimates an initial image that captures rough structure and colors in the missing region. This image is generated by probabilistically estimating the gradient within the missing region based on edge segments intersecting its boundary, and then by flooding the colors on the boundary into the missing region. The color flooding is formulated as an energy minimization problem, and is efficiently optimized by the conjugate gradient method. Finally, by locally replacing the missing region with local patches similar to both the adjacent patches and the initial image, the inpainted image is synthesized. The initial image not only serves as a guide to ensure that the underlying structure is preserved, but also contributes to prune the candidate patches in the patch selection process, which leads to substantial speedup. Experiments show the proposed method outperforms previous methods in terms of both image quality and computational speed, namely more than 5 times faster than the state-of-the-art method.

1. Introduction

Inpainting is a technique for restoring damaged paintings and photographs by filling in missing regions with textures surrounding them [1]. This technique has been used in a variety of applications, e.g., interactive image editing for photo retouching, removing pedestrians from street images for privacy protection [2], and so on.

Various inpainting methods have been proposed to date. One of the most common inpainting techniques is an exemplar-based method, which restores the missing region by pasting square patches sampled from the exterior of the missing region. Exemplar-based methods can be further classified into two categories: greedy and globally optimal methods. Greedy methods iteratively paste patches into the missing region until no missing area is left. This method was first proposed by Harrison [3], and since then, various modifications have been proposed. Criminisi et al. introduced a "priority" measure which specifies which part of the missing region should be processed prior to the others [4]. Other methods improve the accuracy of patch selection by pruning the patch candidates based on image segmentation [5,6]. Although these methods are faster than globally optimal methods, to be described later, they are by nature prone to local minima due to its greedy procedure, often resulting in discontinuity in the inpainted region. An example of this type of error is shown in Fig.1. The region to be inpainted is depicted with red mesh in Fig.1(a), and a close shot of the inpainted result by [4] is shown in Fig.1(b). Discontinuity is apparent on the wall and the window.



(a) original image (b) result by [4] (c) result by [8]
(d) initial image (e) final result

Figure 1. Example of inpainted result by previous and proposed methods

On the other hand, globally optimal methods treat inpainting as combinatorial optimization on patch selection [7-11]. These methods estimate the optimal combination of patches to fill in the missing region by minimizing the overall discontinuity within the region for a given set of patches. Patch selection is optimized by Belief Propagation [7] or EM algorithm [8]. These methods, in contrast to greedy methods, are capable of synthesizing continuous inpainted region. However, optimization process requires far more computation than greedy methods. Another issue is that globally optimal methods do not ensure that the underlying structure in the missing region is preserved. An example for this, obtained by [8], is shown in Fig.1(c). Although a continuous inpainting result is obtained, the wall fades away before reaching the floor, blocked by the windows coming in from its both sides.

Recent advances in inpainting have been focused on tackling these two issues. PatchMatch [10], which is implemented in Photoshop CS5, employs an approximate nearest neighbor algorithm to speed up the patch selection. Although this method runs faster than the previous globally optimal methods, it shares the same problem with them in that the underlying structure is not preserved. Shift-Map [11] is another fast inpainting method, which carries out inpainting by shifting exterior regions into the missing region. Here, the offset with which each region is shifted is constrained to be as uniform as possible, thus each shifted region is expected to preserve the underlying structure. The method, however, often generates discontinuity on boundaries where different regions meet. To summarize, developing an inpainting method that is both structure-preserving and computationally efficient still remains an open problem.

We propose an inpainting method that is capable of preserving the structure underlying the missing region, and is also computationally efficient. In the proposed method, first, an initial inpainted image which captures rough structure and colors in the missing region is esti-

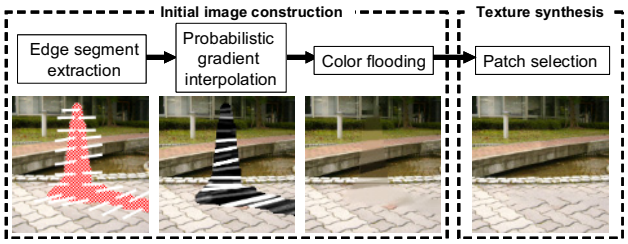


Figure 2. Overview of proposed method

mated, assuming the underlying structure consists of lines and homogeneous regions. The initial image for Fig.1(a) is shown in Fig.1(d). Although this image lacks fine textures in the missing region, the base colors in the region are correctly recovered. As can be seen, this image captures the structure, e.g. straight line between post and floor. Then, using this image as a guide, patches are sampled from the exterior of the missing region and pasted into the region iteratively. The final result is shown in Fig.1(e). As can be seen, structure-preserving inpainted image is obtained by the proposed method. This patch selection guided by the initial image not only ensures to preserve the underlying structure; it also enables the patch selection to be carried out in a greedy manner. This gives us substantial speedup against previous methods, which require optimization over selected patches.

The proposed method was evaluated and compared with the state of the arts, including Photoshop CS5 and Shift-Map [11], and has been proven to outperform them in terms of both image quality and computational speed, namely more than 5 times faster than Shift-Map.

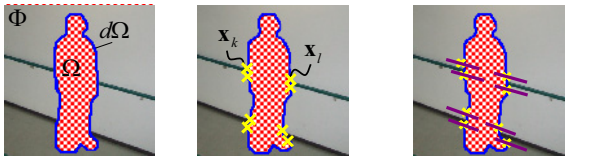
2. Proposed Method

The overview of the proposed method is shown in Fig.2. The proposed method consists of two steps: initial image construction and texture synthesis. In the first step, an initial image, which captures rough structure and colors in the missing region, is estimated. This image is generated by three processes: 1) extracting edge segments intersecting the boundary of the missing region, 2) probabilistically interpolating the gradient inside the region, and 3) flooding colors on the boundary of the region to the topographic relief formed by the gradient magnitude. In the second step, patches are sampled from the exterior of the region and pasted into the region iteratively in a greedy manner. Each step is detailed below.

Note that the proposed method differs from [12] in that the color flooding process is formulated as a continuous optimization problem, instead of discrete optimization. This reformulation leads to speeding up the color flooding process by a factor of 10.

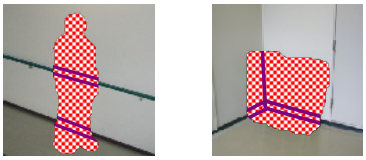
2.1. Initial image construction

2.1.1 Edge segment extraction. An example of an image with a missing region is shown in Fig.3(a). This step starts from extracting edge segments intersecting the boundary $d\Omega$ between the missing region Ω and the source region Φ. First, end points of the edge segments are detected by searching local maxima of gradient along the boundary $d\Omega$ (Fig.3(b)). Then, Hough transform is applied to the source region Φ. Here, votes corresponding to lines that do not pass through the end points are discarded. This allows us to detect only edge segments that



(a) missing region (b) end points (c) edge segments

Figure 3. End points and edge segments extraction



(a) full edges (b) half edges

Figure 4. Two types of edges considered

intersect the end points (Fig.3(c)). Finally, each edge segment detected is associated with its orientation θ_k along with the corresponding end point's position $\mathbf{x}_k$.

2.1.2 Probabilistic gradient interpolation. Next, the gradient inside the missing region is interpolated according to the detected edge segments. Here, the basic idea is to extend each edge segment into the missing region, and to assign the gradient of the edge segment to every pixel on the extended edge. Previous methods extend edge segments by hand [6, 9], or deterministically based on some heuristics [5]. In general, however, it is unknown to what extent each edge should be extended. For example, in Fig.4(a), it is obvious that edge segments on one side of the boundary should be fully extended to the other side, whereas in Fig.4(b), edge segments should be terminated where they intersect with other edges. These examples suggest that 1) if an end point has a corresponding point on the other side of the boundary, the edge segment is likely to be fully extended, and 2) the more intersections an edge encounters, the less likely it is to be further extended.

We employ this probabilistic approach to interpolate the gradient in the missing region. Here, we refer to the former type of edges as full edges, and the latter as half edges. First, for each edge segment detected in the previous process, we consider two hypotheses: one for being a full edge and the other for being a half edge, and for every pixel in the missing region, likelihood for belonging to each hypothetical edge is computed. Finally, each pixel is determined to belong to an edge that gives maximum likelihood.

The former likelihood, i.e. the likelihood that a pixel at $\mathbf{x}$ belongs to a full edge connecting end points $\mathbf{x}_k$ and $\mathbf{x}_l$, is defined such that it gets higher if a pixel $\mathbf{x}$ is close to a hypothetical full edge whose end points $\mathbf{x}_k$ and $\mathbf{x}_l$ have similar color and edge orientation, that is,

$$L_{full}(\mathbf{x},k,l) = L_0 \exp\left[-\frac{p(\mathbf{x}_k,\mathbf{x}_l)^2}{2\sigma_p}\right]\cdot\exp\left[-\frac{d_{kl}^2(\mathbf{x})}{2\sigma_d}\right].$$

Here, $p(\mathbf{x}_k,\mathbf{x}_l)$ is the dissimilarity between $\mathbf{x}_k$ and $\mathbf{x}_l$ in terms of color and orientation, and $d_{kl}(\mathbf{x})$ is the distance from $\mathbf{x}$ to the edge connecting $\mathbf{x}_k$ and $\mathbf{x}_l$. L_0, σ_p and σ_d are parameters determined empirically.

The latter likelihood, i.e. the likelihood that $\mathbf{x}$ belongs to a half edge starting from $\mathbf{x}_k$, is defined such that it gets higher if a pixel $\mathbf{x}$ is close to a hypothetical half edge:

$$L_{half}(\mathbf{x},k)=w^{-n(k,\mathbf{x})}\cdot\exp\left[-\frac{d_k^2(\mathbf{x})}{2\sigma_d}\right],$$

where w is a constant less than 1, $n(k,\mathbf{x})$ is the number of intersections with other edges encountered between $\mathbf{x}_k$ and $\mathbf{x}$, and $d_k(\mathbf{x})$ is the distance from $\mathbf{x}$ to the half edge starting from an end point $\mathbf{x}_k$. Note that each time the edge intersects another, this likelihood diminishes by a factor of w. This models the nature of a half edge that it is less likely to be extended past the intersections to other edges. After computing likelihoods for each hypothetical edge, the gradient $\nabla\tilde{I}(\mathbf{x})$ at pixel $\mathbf{x}$ is given as that of the edge with the maximum likelihood, weighted by the likelihood:

$$\nabla\tilde{I}(\mathbf{x})=\begin{cases}\nabla I_{\mathbf{x}}^{full}\ L_{full}\left(\mathbf{x},k_{\mathbf{x}}^{full},l_{\mathbf{x}}^{full}\right) & if\ \ L_{full}\left(\mathbf{x},k_{\mathbf{x}}^{full},l_{\mathbf{x}}^{full}\right)\geq L_{half}\left(\mathbf{x},k_{\mathbf{x}}^{half}\right)\\ \nabla I_{\mathbf{x}}^{half}\ L_{half}\left(\mathbf{x},k_{\mathbf{x}}^{half}\right) & otherwise\end{cases}$$

where $\nabla I_{\mathbf{x}}^{full}$ is the average of intensity gradient vectors at $\mathbf{x}_k$ and $\mathbf{x}_l$, and $\nabla I_{\mathbf{x}}^{half}$ is the gradient at $\mathbf{x}_k$.

2.1.3 Color flooding. An initial image is generated by flooding colors on the boundary $d\Omega$ to the topographic relief formed by the gradient magnitude. We formulate this as a continuous optimization problem, and obtain the optimal solution using the conjugate gradient method.

We consider a regular grid that covers the entire missing region and its neighbor in the original image, Here, each node corresponds to a pixel in the image. The energy function to be minimized is defined as follows:

$$F(\mathbf{X})=F_{data}(\mathbf{X})+F_{smooth}(\mathbf{X}),$$

where $\mathbf{X}=(X_1,\cdots,X_p,\cdots,X_N)$ is a set of colors for each pixel. The data term in the above energy is defined as

$$F_{data}(\mathbf{X})=\sum_{p\in\Phi}\left\|X_p-T(\mathbf{x}_p)\right\|^2,$$

where $T(\mathbf{x}_p)$ is the color at $\mathbf{x}_p$ in the original image. This term penalizes the estimated colors deviating from its original colors. The smoothness term is given by

$$F_{smooth}(\mathbf{X})=\sum_{p,q\in R}V_{pq}\cdot\left\|X_p-X_q\right\|^2,$$

where R is a set of adjacent nodes. V_{pq} is a weight for the smoothness term, which is designed such that the closer the direction from $\mathbf{x}_p$ to $\mathbf{x}_q$ is to the gradients at $\mathbf{x}_p$ and $\mathbf{x}_q$, the lighter it gets, i.e.,

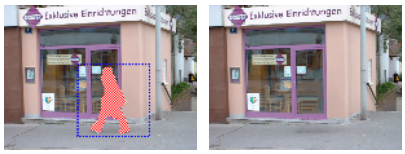
$$V_{pq}=\exp\left[-\alpha\left|\mathbf{n}_{pq}\cdot\left(\nabla\tilde{I}(\mathbf{x}_p)+\nabla\tilde{I}(\mathbf{x}_q)\right)\right|\right]$$

where $\mathbf{n}_{pq}$ is the unit vector whose direction coincides with that from $\mathbf{x}_p$ to $\mathbf{x}_q$, $\cdot$ denotes the inner product of vectors, and α is a parameter determined empirically. This weight adds an edge-preserving characteristic to the smoothness term, where abrupt color transition is tolerated along the gradient, i.e. across the edge, while imposing color uniformity to the other directions.

2.2. Patch selection

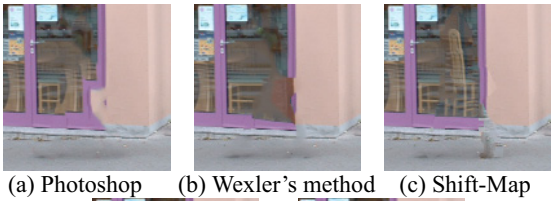
In the second step, the final inpainting image is synthesized by sampling patches from the source region Φ and by pasting them into the missing region Ω iteratively. Here, a patch $\Psi(\mathbf{x}_j)$, i.e. a patch centered at $\mathbf{x}_j$ in the source region Φ, is selected as a patch to be pasted to $\mathbf{x}_i$ according to the following criteria: 1) a patch $\Psi(\mathbf{x}_j)$ is continuous to the already pasted patches in the neighbors of $\mathbf{x}_i$, and 2) is similar to corresponding region in the initial image. Formally,

$$\Psi(\mathbf{x}_j)=\arg\min_{\mathbf{x}_j\in\Phi}c_{adj}\left(\hat{\Psi}(\mathbf{x}_i),\Psi(\mathbf{x}_j)\right)+c_{init}\left(\tilde{\Psi}(\mathbf{x}_i),\Psi(\mathbf{x}_j)\right),$$

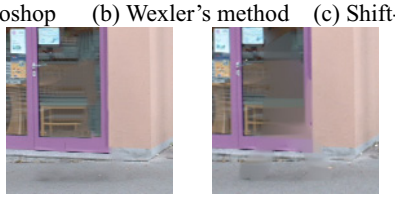


(a) original image (b) inpainted result

Figure 5. Experimental result on house image



(a) Photoshop (b) Wexler's method (c) Shift-Map



(d) proposed method (e) initial image

Figure 6. Comparison of inpainted results

(a) original image (b) inpainted result (c) initial image

Figure 7. Experimental results on natural images

where $\hat{\Psi}(\mathbf{x}_i)$ and $\tilde{\Psi}(\mathbf{x}_i)$ are the patches at $\mathbf{x}_j$ in the inpainted image under construction and in the initial image, and $c_{adj}(\Psi(\mathbf{x}_a),\Psi(\mathbf{x}_b))$ and $c_{init}(\Psi(\mathbf{x}_a),\Psi(\mathbf{x}_b))$ are the sum of squared differences (SSD) of two patches $\Psi(\mathbf{x}_a),\Psi(\mathbf{x}_b)$. Note that in $c_{adj}(\Psi(\mathbf{x}_a),\Psi(\mathbf{x}_b))$, difference is computed only at pixels already filled by previously pasted patches. Patch selection guided by the initial image not only ensures to preserve the underlying structure, it also allows the patch selection to be carried out in a greedy manner. This gives us substantial speedup against previous methods, which require optimization over selected patches.

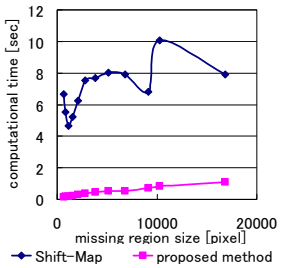
3. Experiments

3.1. Inpainted results

To demonstrate the performance of our method, we performed experiments on real street images with pedestrians. Parameters used in the following experiments are $L_0=9$, $\sigma_p=6\times10^2$, $\sigma_p=0.5$, $w=0.5$ and $\alpha=2$, respectively. An experimental result is shown in Fig.5. Here, the original image with a missing region and the inpainted result by the proposed method are shown in Fig.5 (a) and (b), respectively. Close shots of the results obtained by Photoshop CS5, Wexler's method [8], Shift-Map [11] and the proposed method are compared in Fig.6 (a), (b), (c) and (d), respectively. The initial image of the proposed method is also shown in Fig.6 (e). As can be seen, none of the previous method succeeded in reconstructing the underlying structure: the contour of the door is heavily distorted (Fig.6 (a)), or the contour fades away in the middle (Fig.6 (b) and (c)). On the other hand, as shown in Fig.6 (d), the proposed method successfully recovers the frame of the door. This is because

(a) original image (b) inpainted result (c) initial image

Figure 8. Experimental results on barrel image



(a)

Process	Running Time [sec]
Edge segment extraction	0.058
Gradient interpolation	0.035
Color flooding	0.21
Patch selection	0.54
Overall	0.84

(b)

Figure 9. Evaluation of computational time

the structure of the door is roughly reconstructed in the initial image (Fig. 6 (e)), and this image is utilized as a guide in generating the final result (Fig.6 (d)).

Although the proposed method assumes the missing region to be composed of linear structures, it also works with images with structures that are not completely linear, but can be approximated to be linear. An experimental result for this type of image is shown in Fig.7. The missing region, the inpainted result and the initial image are shown in (a), (b) and (c), respectively. In the initial image, spurious linear edges are reconstructed. In the inpainted result, however, these edges are correctly replaced with textured regions. This result shows that the initial image does not need to be precise, i.e. approximate reconstruction is sufficient, since it is only used as a guide in the later patch selection process.

The experimental result in Fig.8 shows the limitation of our approach. The proposed method has failed in reconstructing the rim of the barrel. The reason for this failure can be summarized into two points: 1) in pasting patches of the rim, they need to be rotated in accordance with the normal of the rim, and 2) a guide is needed that forces patch to be aligned on an arc. Although the proposed method guides the patch pasting process by means of an initial image, it assumes the underlying structure to be composed of lines. For nonlinear structures, the method would fail in constructing the initial image, as shown in Fig.8(c). Extending the method to deal with more general types of structures remains future work.

3.2. Computational time

Finally, we evaluated the computational time for the proposed method and compared with that of the Shift-Map [11], which is one of the fastest globally optimal methods. Since the computational time is expected to depend on the size of the missing region, we measured the time for images with different missing region size, ranging from approximately 500 to 17,000 pixels. These images were generated by resizing the same image with the same missing region at different magnification ratio, and then by clipping them into 480×360 pixel images with the missing region centered. All the images were processed on a PC with 2.7GHz CPU and 4.0 GB RAM.

The computational time for the proposed method and Shift-map are shown in Fig.9(a). As can be seen, the proposed method is more than 5 times faster than Shift-Map for any missing region sizes. It should also be noted how steadily the computational time for the proposed method increases, whereas that for Shift-Map increases quite unstably.

Further analysis on the computational time for the proposed method is shown in Fig.9 (b). This table shows the computational time for each process in the proposed method, measured on an image with a missing region size of approximately 10,000 pixels. Although the computational time for the patch selection is dominant, it only takes 0.54 sec, making the overall time less than a second. This result shows that pruning candidate patches using the initial image works effectively to speed up the patch selection process.

4. Conclusion

A novel inpainting method based on structure estimation has been proposed. Experimental results show the proposed method is capable of preserving the underlying structure in the missing region, while achieving more than 5 times faster computational speed than the state-of-the-art inpainting method. Currently, the method only deals with images where the structure in the missing region is composed of linear edges. Extending the method to deal with more general type of structures remains our future work.

Acknowledgment: This work was supported by Ministry of Internal Affairs and Communications of Japan.

References

[1] Bertalmio, et al.: “Image inpainting,” *ACM SIGGRAPH*, pp.417-424, 2000.

[2] Arai, et al.: “Pano UMECHIKA: A crowded underground city panoramic view system,”*DCAI2010*, pp.174-181, 2010.

[3] Harrison.:“A Non-hierarchical procedure for re-synthesis of complex texture,” *Proc.WSCG01*, pp. 190-197, 2001

[4] Criminisi, et al.: “Region filling and object removal by exemplar-based image inpainting,” *IEEE Trans. IP* vol.13, no.9, pp.1200-1212, 2004.

[5] Jia, et al.: “Image repairing: robust image synthesis by adaptive ND tensor voting,” *Proc.CVPR*, pp.643-650, 2003.

[6] Sun, et al.: “Image completion with structure propagation,” *ACM SIGGRAPH*, pp.861-868, 2005.

[7] Komodakis, et al.: “Image completion using efficient belief propagation via priority scheduling and dynamic pruning,” *IEEE Trans. IP*, vol.16, no.11, pp.2649-2661, 2007.

[8] Wexler, et al: “Space-time completion of video,” *IEEE Trans. PAMI*, vol.29, no.3, pp.463-476, 2007.

[9] Kawai, et al: “Image inpainting considering brightness change and spatial locality of textures”, *Proc. VISAPP*, Vol.1, pp. 66–73, 2008.

[10] Barnes, et al.: “Patchmatch: A randomized correspondence algorithm for structural image editing,” *ACM SIGGRAPH*, vol.28, no.3, 2009.

[11] Pritch, et al: “Shift-map image editing,” *Proc. ICCV*, pp.151-158, 2009.

[12] T. Shibata, et al: “Image inpainting based on probabilistic structure estimation” *Proc. ACCV*, pp.109-120, 2010.