International Journal of Multimedia and Ubiquitous Engineering
Vol.10, No.4 (2015), pp.193-204
http://dx.doi.org/10.14257/ijmue.2015.10.4.19

Image Completion Using Similarity Analysis and Transformation

Mang Xiao[1], Guangyao Li[1], Yunlan Tan[1,2] and Jie Qin[1]

[1]*College of Electronic and Information Engineering, Tongji University, Shanghai 201804, China*

[2]*School of Electronic and Information Engineering, Jinggangshan University, Ji'an, Jiangxi 343009, China)*

fathomguru@163.com

Abstract

Image completion is aim to fill the missing regions in images. A robust completion technique using similarity analysis and transformation is proposed to address this problem. Firstly, in order to decrease the search space of patches, random mapping method is used to analyze texture regions which have similar structure and texture with damaged regions. Secondly, geometric and photometric transformations of image are adopted to find the best patches. Thirdly, increasing the accuracy of the structure propagation, a priority calculation method is optimized based on confidence factor and edge information. Finally, a number of examples on real and synthetic images show the effectiveness of our algorithm for image completion.

Keywords: *Image completion; similarity analysis; approximate nearest neighbor; texture synthesis*

1. Introduction

Image completion is generally applied to automatically filling in the missing regions of an image in a visually reasonable way. In recent years, it has been one of the most important techniques in the field of image processing and computer vision.

Image completion [1] techniques can be divided into two main categories: structure-based and texture-based.

The structure-based [2] methods complete the missing regions using thermal diffusion equations which can propagate the image information of the regions around the damaged ones. These methods include TV (Total Variation) model [3-4], Euler's elastic model [5-7], Mumford-Shah model [8] and so on. They perform well in completing small missing areas of the picture while is prone to blurring in large ones.

The texture-based methods [9-15] extract the texture features of the intact regions of the damaged picture and use them to composite new image patch, in order to eliminate the blurring in large damaged regions restoration. However, this kind of techniques usually misses the structure information of images. To solve this problem, Criminisi *et al.* [16, 17] were inspired by the artificial repair methods. They calculated the priorities of the pixels around the damaged area according to certain rules separately, and used them to decide the order of the completion.

This technique has significantly improved the quality of the restoration, but it still exists some defects: On one hand, the algorithm is susceptible to the confidence factors during the structure propagation.

Cheng *et al.* [18] found out the problem that the value of filling priority often decreased to zero quickly, and introduced a robust priority calculation algorithm which raised both the robustness and repair effects a lot but causing high time complexity. Further improvements modify the search and sampling approaches for better structure

ISSN: 1975-0080 IJMUE

preservation [19-23]. Patch optimization based approaches have now become common practice in texture synthesis [24-29].

On the other hand, damaged image is usually a large region while the sample patches are very small, so it takes plenty of time to search for similar sample patches in the entire image. Many solutions have been proposed to reduce the time consumption.

Ashikhmin [30] proposed a local propagation technique exploiting local coherence in the synthesis process by limiting the search space for a patch to the source location of its neighbors in the exemplar texture. Tong *et al.* [31] improved Ashikhmin's method and introduced the k-coherence technique. He *et al.* [32] used kd-trees to accelerate the speed of texture synthesis. To avoid the local minimum trap and the quantity limitation of the k-coherence, Barnes *et al.* [33-34] regarded amount of pixels as center patches to select the match-patches randomly, find out similar ones quickly and transform the location information of the similar ones to adjacent pixels. Then the processes were repeated again and again until the optimal sample patches were found. Unfortunately, Barnes's method may cause local discontinuity of the image and is still not perform well in structure propagation.

A robust completion technique using similarity analysis and transformation is proposed by according to the above researches. Firstly, in order to decrease the search space of samples, random mapping method is used to analyze texture regions which have similar structure and texture with damage regions. Secondly, geometric and photometric transformations of image are adopted to find the best patches. Thirdly, increasing the accuracy of the structure propagation, a priority calculation method is optimized based on confidence factor and edge information. Finally, some experiments will be done to show the high efficiency and good quality of our method.

2. Prerequisite Knowledge of Image Completion

This section provides a brief description of the image completion. The source region of the damaged picture is indicated by Φ, the damaged region is denoted Ω and its contour is denoted δΩ. *p* is the pixel on δΩ. Patch Ψ_P is a square with side length *w*, and the center of Ψ_P is *p*. The best match patch Ψ_q from the source region Φ is most similar to those parts that are already filled in Ψ_P.

The priority P(p) of patch Ψ_P is defined as the product of two terms:

$$P(p)= D(p)\, C(p) \tag{1}$$

The D(p) is data term and C(p) is confidence term, and they are defined as follows:

$$D(p)=\frac{|\nabla \mathrm{I}_p^{\perp} * n_p|}{\alpha} \tag{2}$$

$$C(p)=\frac{\sum_{q\in\psi_p\cap(\mathrm{I}-\Omega)} C(q)}{|\psi_p|} \tag{3}$$

Where α is a normalization factor, n_p is a unit vector orthogonal to the front δΩ in the point p and $\perp$ denotes the orthogonal operator. $|\Psi_P|$ is the area of Ψ_P. During initialization, the function C(p) is set to C(p) = 0 , $\forall p \in \Omega$, and $\forall p \in \mathrm{I}-\Omega$, C(q)=1. Find the best matching patch Ψ_q of Ψ_P , copy the pixels in Ψ_q to fill the missing pixels in Ψ_P and update confidence term.

$$C(p) = C(q) \qquad \forall p \in \Psi q \cap \Omega \tag{4}$$

There exist three main problems in this greedy algorithm.

(1) It costs large amounts of time to search the best matching patch in the full image.

(2) Large errors may be caused by only using translation patches to find the best patches.

(3) It is too rough using simple method of SSD (Sum of Squared Differences) to measure the similarity between Ψ_P and Ψ_q .

$$d(\psi_p, \psi_q) = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{m}(p_{ij}^R - q_{ij}^R)^2 + (p_{ij}^G - q_{ij}^G)^2 + (p_{ij}^B - q_{ij}^B)^2} \tag{5}$$

$p_{ij}^R, p_{ij}^G, p_{ij}^B, q_{ij}^R, q_{ij}^G, q_{ij}^B$ respectively denote the pixels value of red, green and blue in Ψ_p and Ψ_q. The similarity between patches are measured by the value of *d*, which the most similarity has minimum value of d.

$$\Psi_{q^\wedge} = \arg\min_{\Psi_q \in \Phi} d(\psi_p, \psi_q) \tag{6}$$

It's hard to find the optimal patch, while having more than one patch of the same least value of d.

3. Our Algorithm

This paper introduces a new algorithm to improve the Criminisi's method. Considering the image feature distributes sparsely, a randomized correspondence algorithm is adopted to analyze the similarity between target regions and source regions, which having the similar structure and texture. So we can reduce the redundant searching space to speed up the algorithm. On the other hand, we transform the patches on geometric and photometric, optimizing the method to search best patches based on confidence factor and edge information, and optimizing the method to calculate the most similarly patches. Therefore, we can use these measures to improve image completion that get more wonderful results.

3.1. Similarity Analysis

In order to speed up the processing of image completion, it is important to reduce the time of searching the best matching patch. Therefore, it is critical to use similarity analysis to reduce the searching regions.

As defined in Section 2, adjacent pixels in contours can be effectively propagate information, that randomized correspondence method can be used to find nearest-neighbor for every pixel.

A nearest-neighbor field (NNF) function is defined as f: δΩ --> Φ. Given patch coordinate (x, y) in δΩ and its corresponding nearest neighbor (x', y') in source region Φ, set f(x, y)=(x', y'), and these values are stored in an array.

There has three main components of the random mapping algorithm, including initiation, propagation and random search, as shown in Figure 1.

(1) Initiation: We initialize the nearest neighbor field by assigning random values. As shown in Figure 1(a), corresponding to three random given dashed-line patches

distributed in different locations, there has three adjacent solid-line patches that center pixel is in the contour.

The iteration process is aim to improve the similarity with patches. According a fixed scan order, each iteration is to find the best match-patch. The odd-order scan is from right to left, bottom to up and the even-order is from left to right, top to bottom. For example, V_i and W_i are defined, respectively, propagation and random search at patch i, we will proceed in the order: V_1, W_1, V_2, W_2, ⋯, V_n, W_n.

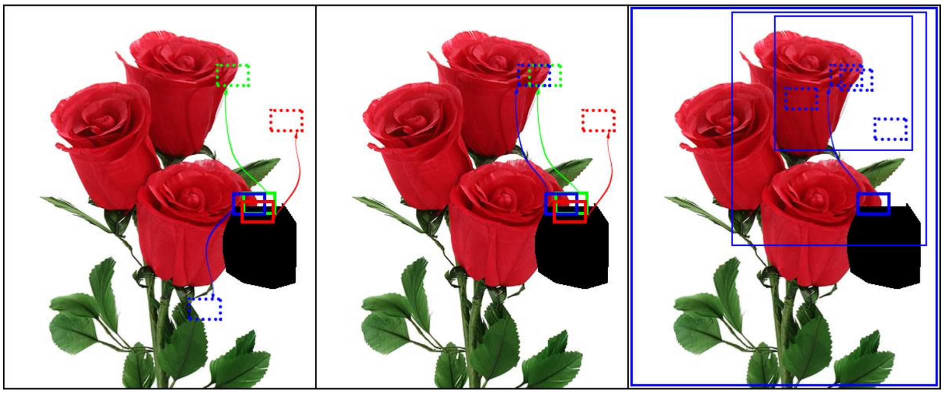
(2) Propagation：Searching the similar patch f(x, y) to the patch w with center at (*x*, *y*), we can use the coordinates of similar patches *f*(*x*-1, *y*) and *f*(*x*, *y*-1). The SSD value between patch w and patch v is denoted D(w, v). So the most similar patch to patch w have the least value in {D(w,f(x,y)), D(w, f(x-1,y)), D(w,f(x,y-1))}, then assign coordinate of this patch to f(x y).

This method's main advantage is that just some patches correctly map to similar patches in a continuous area, then other patches can obtain the corresponding high similar patches with information propagated through these patches. As shown in Figure 1(b), when look for the similar patch of the blue patch, firstly verify whether the red patch has a correct mapping, if it has correct one, then we can fast search a good similar patch but searching all patches in whole image.

(3) Random search：As shown in Figure 1(c), for the purpose to improve the similar accuracy of patches, we define a coordinate mapping function to search the similar patch in a large region centered f(x, y).

$$z_i = f(\mathrm{x}, \mathrm{y}) + r\eta^i R_i \tag{7}$$

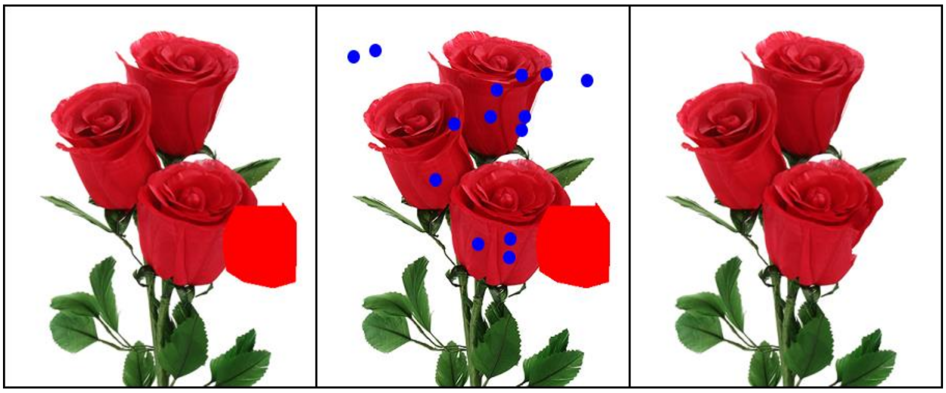
Where *r* is a large maximum search radius, η is a fixed ratio between search window sizes, and $\mathrm{R_i}$ is a uniform random in[-1,1]×[-1,1]. We find patches for i=0, 1, 2, … until the search radius $r\eta^i$ is less than 1. Empirical value η=0.5.



(a) (b) (c)

Figure 1. Random Mapping Algorithm

Image is divided into many overlap cells with size of m*m pixels, and the number of every mapped cell is counted to look for the best match-patch, then top k cells of largest mapping number are chose. As shown in Figure 2(b), these k blue points are the centers of top k cells of the Figure 2(a). The result of image completion is shown in Figure 2(c). We set value of k ranges from 10 to 15. As shown in Figure 3, the center of top 14 cells of highest mapping number are counted of Figure 2(b)(330×250-pixel image), here m=60.



(a) (b) (c)

Figure 2. Photograph of a Rose: (a) Damaged Image; (b) The Blue Points are the Centers of Top k Cells having Largest Mapping Number; (c) Restored Image with our Algorithm

As shown in in Figure 2(b), the blue points are centers of final searching locations, which include almost all the main structure and texture. Therefore, only searching these main regions can make image completion well and fast. We can see the caparison of searching number of exemplars in our and Criminisi's algorithm from Figure 4.

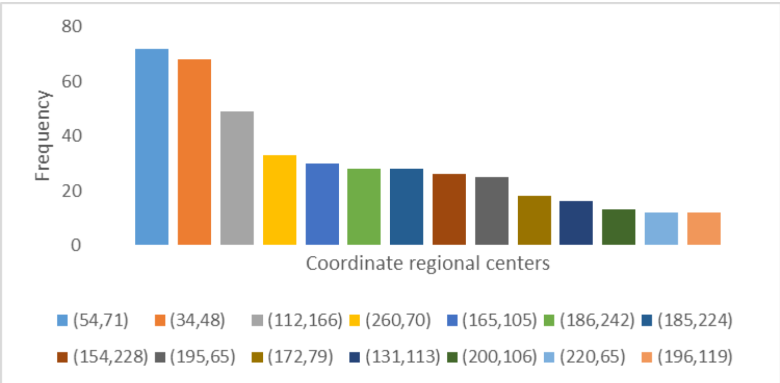


Figure 3. The Statistics of Cells Mapped

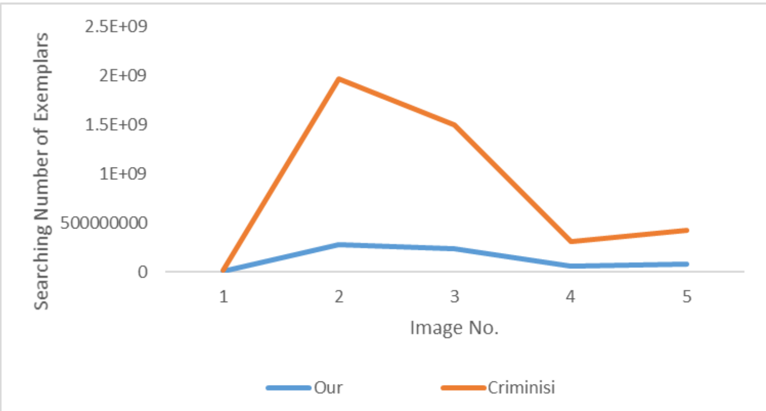


Figure 4. The Comparison of Searching Number of Exemplars in our and Criminisi's Algorithm

3.2. Transformation

Since the target contour is reconstructed with geometry changing, it is difficult to find good source patches for target patches by simple translation. Therefore, for a transform parametrised by a set θ, the source patches s_i are transformed to $S_{\theta_i} = \{ f_{\theta_i}(\mathrm{S}(\mathrm{g}_{\theta_i}(\mathrm{j}_p)))\}$, $p \in \{1...\mathrm{P}\}$ where g_{θ_i} is the geometric transformation function working on patch indices and f_{θ_i} is the photometric transformation function working on pixel color. Each patch contain P pixels. The relative position of each pixel in the patch is given by j_p for all $p \in \{1...\mathrm{P}\}$, for example a patch centered at i contain pixels i+ j_p .

The geometric transformation we consider is

$$\mathrm{g}_{\theta_i}(\mathrm{j}_p) = \alpha_i R_{-\phi_i}\, \mathrm{j}_p + x_i \tag{8}$$

Where α_i is the scale, ϕ_i is the rotation, j_p is a pixel in a patch around i, and x_i is the translation, as shown in Figure 5. A rotation ϕ_i of the source image is equivalent to rotation $-\phi_i$ of the patch indices, so the indices are rotated by rotation matrix $R_{-\phi_i}$. To make optimization easier, we limit the range of scale to $1 \le \alpha_i \le 2$. These parameters are part of the parameter set θ.

We consider the following photometric transformation

$$f_{\theta_i}(\mathrm{s}) = \mathrm{s} + \beta_i \tag{9}$$

Where s denote the value of one color channel of a pixel, β_i is the shift of brightness that completes the parameter set θ. We can simply turn off a transformation by setting the parameter to its default value: α_i =1, ϕ_i =0, β_i =0.

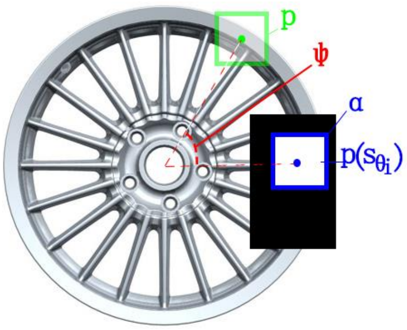


Figure 5. Geometric Transformation

3.3. Patch Priorities

The robust priority function is defined with image transformation as follows:

$$P(p) = w_c \bullet G_c(p(\mathrm{s}_{\theta_i})) + w_d \bullet D(p(\mathrm{s}_{\theta_i})), \quad 0 \le w_c, w_d \le 1 \tag{10}$$

Where w_c and w_d are respectively the component weights of the confidence and the data terms. Note that $w_c+w_d=1$. In order to to smooth the decreasing curve of the confidence term to match that of the data term, $G_c(p)$ is defined a regularizing function

$$G_c(p) = (1-h) \times C(p) + h, \quad 0 \le h \le 1 \tag{11}$$

ISSN: 1975-0080 IJMUE

Where h is regularizing factor to control the decreasing of curve smoothness. Without loss of generality, h is empirically set to 0.7. Therefor the value range of $G_c(p)$ is regularized to [h, 1].

3.4. Searching Best Match-patch

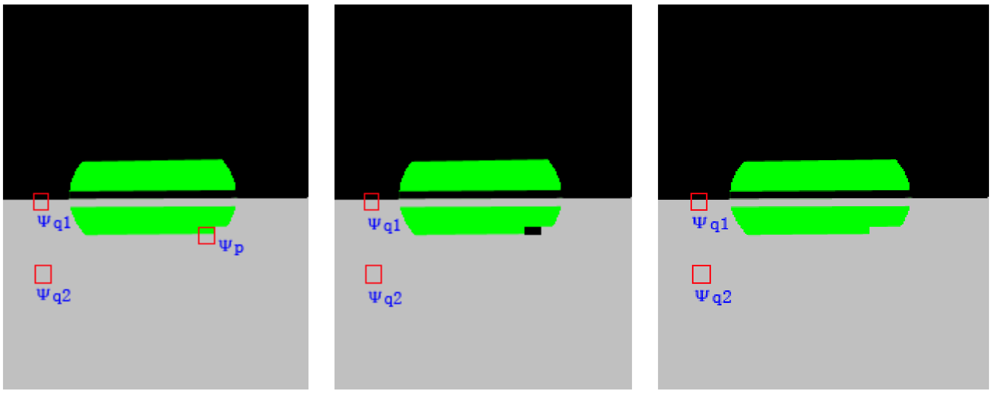
In generally, as there could has more than one patches that have the same least SSD value, it may result in accumulation of error if don't choose the best match-patch. Therefore we need to compare the variance of these patches for choosing the best match-patch. In order to calculate the pixels mean both known area p of filling patch and corresponding area of match-patch, the function is defined as follows.

$$M = \frac{\sum l_{p \in \Phi \cap \Psi}}{\#\{p \mid p \in \Phi \cap \Psi\}} \tag{12}$$

Where # denotes the number of pixels and ι is pixel value. The function to calculate the variance of match-patch area that correspond to the filling patch unknown area q is defined as follows.

$$S = \frac{\sum (l_{q \in \Phi - \Psi} - M)^2}{\#\{q \mid q \in \Phi - \Psi\}} \tag{13}$$

As shown in the Figure 6(a), the target patch Ψ_P has two best match-patch Ψ_{q1} and Ψ_{q2} with same SSD value. If formula (6) was taken to choose the best match-patch, then it could choose patch Ψ_{q1} to fill patch Ψ_P and made the bad result as shown in Figure 6(b). Else if we took the improved formula (13) to choose the best match-patch, we could sure choose patch Ψ_{q2} to fill patch Ψ_P and made the better result as shown in Figure 6(c).



(a) Searching match-patch (b) Error match-patch (c) Best match-patch

Figure 6. Choose Best Match-patch

4. Experimental Results and Analysis

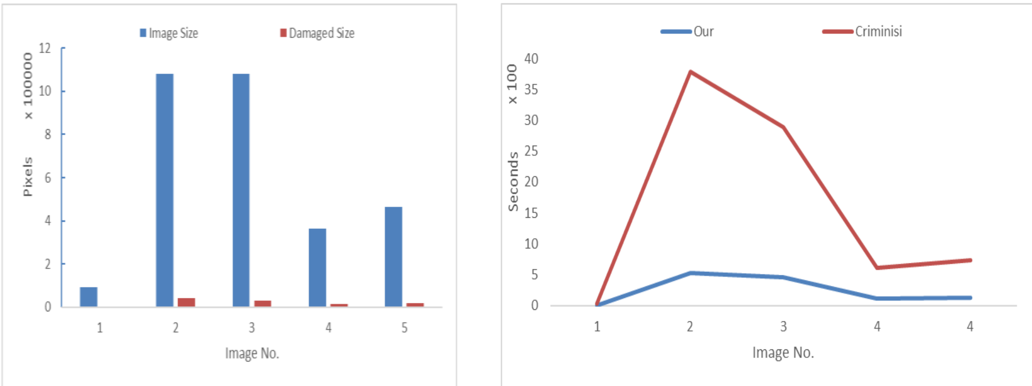
In order to test the effectiveness of the proposed algorithm, some experiments are produced and compared to the so-obtained results with conventional methods. All experiments are run on an Intel dual-core 2.8GHZ with 4GB of RAM.

These selected images for testing are from both natural and synthetic images. As shown in Figure 9-13, the results of the Criminisi's repair algorithm that have many discontinuous areas. The result in last row even appears much leaves texture in flower, and it may be a big mistake in human eyes. With raising the number of iterations for image completion, the confidence value of filling patch becomes smaller that increase the probability of error, which could result in drawing incorrect structure and texture. The results of Wexler's [11] approach are also as blur as ones in all rows. The mainly reasons were that it is not always able to find the global optimum solution via optimized the global energy function and the pixels values may be obtained from the weighted neighbor pixel values. While the calculation method of patches priority is optimized in our algorithm, which improve the technique of searching the best match-patch and reduce the searching regions of exemplar. Therefore we can achieve better visual effect by greatly speeding up the image completion and decreasing the error rate of image completion.

As shown in Figure 2(b), the size of all the main regions is very small, and the total number of 9*9 pixels patches in the main regions is about 10% of the whole image. Therefore, it is very effect to reduce the completion time of the High Definition (HD) image, overall, we can save nearly 81% time without considering the time cost of initializing partial exemplar regions.

As shown in Figure 4, the number of searching patches of Figure 9-13 by our algorithm is compared with Criminisi's algorithm. It is easy to see that the time of our algorithm grows more slowly as the increasing of image size.

It is the size of image and damaged regions in Figure 7(a). The time consumed by our algorithm is compared with Criminisi's algorithm, as shown in Figure 7(b). The data indicate that our algorithm saves more time of HD image than small image.



(a) Size of image and damaged regions (b) Comparison of running time

Figure 7. Comparison of Criminisi's Algorithm and our Algorithm

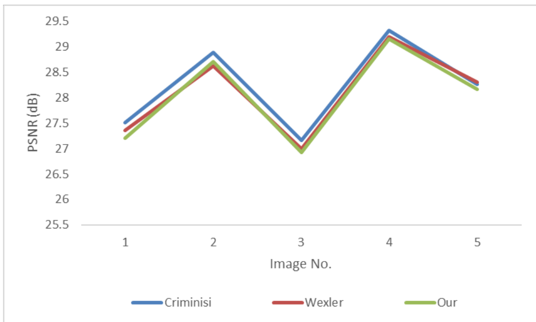
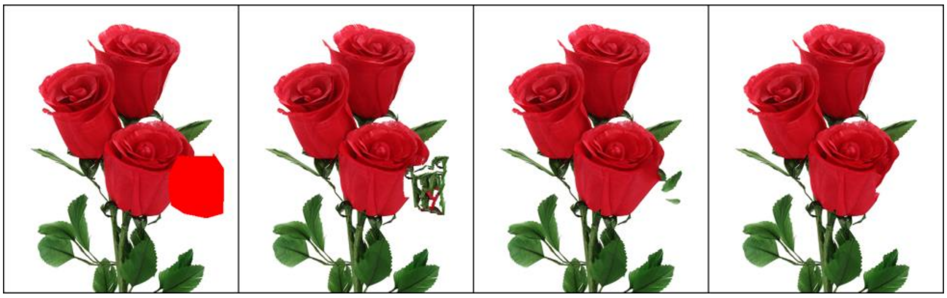


Figure 8. Image Quality Assessment with PSNR

With the purpose of evaluating the impletion quality of our algorithm, Peak Signal to Noise Ratio (PSNR) is used to detect the image completion quality of Figure 9-13. As shown in Figure 8, PSNR1, PSNR2, PSNR3 respectively denote the PSNR values of the image completion by Criminisi, Wexler and our algorithm. The results show that the repaired images by our algorithm are better than other two algorithms in image consistency and coherence judging from human eyes. However, these advantageous do not show up very well in PSNR that is different from human visual system. Therefore it is more reliable to verify the effect with the human visual system.



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

Figure 9

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

Figure 10

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

Figure 11

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

Figure 12

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

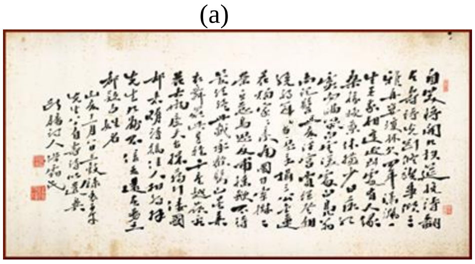
Figure 13

Figure 7 some results. The first column shows unknown regions (red). The column (a) to column (d) are respectively denote the results completed by Criminisi, Wexler and our algorithm.

Further Examples on Photograph. We show more examples on photographs of real scenes. Figure 14 shows the result of our algorithm on region filling of a Chinese text image. Figure 14(a) is original photo of type written Chinese text. Figure 14(b) is a portion of the image (marked with a red region) that has been removed. Figure 14(c) is the result of our filling algorithm. Figure 14(d) show the detail of text synthesized in the target region highlighted. Even though the generated Chinese text does not make much sense, it still looks plausible. This example further demonstrates the advantage of our algorithm in the case of pure texture synthesis.


(a)


(b)


(c)

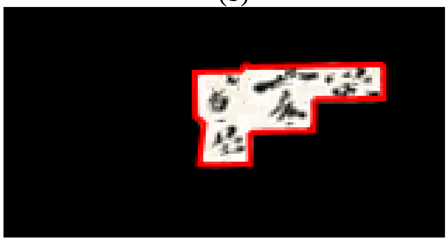

(d)

Figure 14. Examples on Chinese Text Photograph

Another Example of Image Completion. (a) Original image. The text occupies 8.5% of the total image area. (b) Result of text removal via our algorithm.


(a) Original image


(b) Result of our algorithm

Figure 15. Removing the Text from Image

5. Conclusions

A fast image completion algorithm using similarity analysis and transformation is proposed, which can quickly repair the damaged image. There has three main contribution of this work. Firstly, similarity analysis is used to find the minimal regions including main structure and texture. Secondly, geometric and photometric transformations of image are adopted to find the best patches. Thirdly, increasing the accuracy of the structure propagation, a priority calculation method is optimized based on confidence factor and edge information. Experimental results show that our algorithm is effective both in the speed and the visual quality. However, there still has a fault of greedy algorithms that once a wrong patch selected could accumulate small errors to a serious error in the end. Therefore, how to use some other auxiliary information to reduce the error rate could be further research.

Acknowledgements

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

References

[1] C. Guillemot and O. L. Meur, “Image inpainting: Overview and recent advances”, Signal Processing Magazine, vol. 31, no.1, **(2013).**

[2] M. Bertalmio, G. Sapiro, V. Caselles and C. Ballester, “Image inpainting”, Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, **(2000)** 417-424, New Orleans, Louisiana, USA.

[3] T. Chan and J. Shen, “Local inpainting models and tv inpainting”, SIAM J. Appl. Math, vol. 62, no. 3, **(2001).**

[4] P. Arias, G. Facciolo, V. Caselles and G. Sapiro, “A variational framework for exemplar-based image inpainting”, International journal of computer vision, vol. 93, no. 3, **(2011).**

[5] T. F. Chan and J. Shen, “Nontexture inpainting by curvature-driven diffusions”, Journal of Visual Communication and Image Representation, vol. 93, no. 4, **(2001).**

[6] T. F. Chan, S. H. Kang and J. Shen, “Euler's elastica and curvature-based inpainting”, SIAM Journal on Applied Mathematics, **(2002),** pp. 564-592**.**

[7] D. Tschumperlé, “Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's” International Journal of Computer Vision, vol. 68, no. 1, **(2006).**

[8] A. Tsai, J. A. Yezzi and A. S. Willsky, “Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification”, Image Processing, vol. 10, no. 8, **(2001).**

[9] Efros and T. K. Leung, “Texture synthesis by non-parametric sampling”, IEEE International Conference on Computer Vision, **(1999)** pp. 1033-1038, Greece.

[10] Drori, D. Cohen-Or and H. Yeshurun, ”Fragment-based image completion », ACM Transactions on Graphics, ACM, vol. 22, no. 3, **(2003).**

[11] Jia and C Tang, “Image repairing: Robust image synthesis by adaptive nd tensor voting”, Computer Vision and Pattern Recognition, Proceedings. IEEE Computer Society Conference on. IEEE, **(2003)** pp. 643 -650.

[12] Jia and C Tang, “Inference of segmented color and texture description by tensor voting”, Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 26, no. 6, **(2004).**

[13] D. Simakov, Y. Caspi, E. Shechtman and M. Irani, “Summarizing visual data using bidirectional similarity”, Computer Vision and Pattern Recognition, IEEE Conference on. IEEE, **(2008),** pp. 1-8.

[14] Y. Pritch, E. Kav-Venaki and S. Peleg, “Shift-map image editing”, Computer Vision, IEEE 12th International Conference on. IEEE, **(2009)** 151-158.

[15] J. Aujol, S. Ladjal and S. Masnou, “Exemplar-based inpainting from a variational point of view”, SIAM Journal on Mathematical Analysis, vol. 42, no. 3, **(2010).**

[16] A. dfgvdfCriminisi, P. Perez and K. Toyama, “Object removal by exemplar-based inpainting”, Computer Vision and Pattern Recognition, Proceedings. IEEE Computer Society Conference on. IEEE, **(2003)**, pp. 721 -728.

[17] Criminisi, P. Perez and K. Toyama, “Region filling and object removal by exemplar-based image inpainting”, IEEE Transactions on Image Processing, vol. 13, no. 9, **(2004).**

[18] W. H. Cheng, C. W. Hsieh, S. K. Lin, C. W. Wang and J. L. Wu, “Robust Algorithm for Exemplar-Based Image Inpainting”, The International Conference on Computer Graphics, Imaging and Vision. (**2005**), pp. 64-69, Beijing, China.

[19] V. G. Kim, Y. Lipman and T. A. Funkhouser, “Symmetry-guided texture synthesis and manipulation”, ACM Trans. Graph., vol. 31, no. 3, **(2012).**

[20] Galerne, Y. Gousseau and J. M. Morel, “Random phase textures: Theory and synthesis”, Image Processing, IEEE Transactions on, vol. 20, no. 1, **(2011).**

[21] Bugeau, M. Bertalmío, V. Caselles and G. Sapiro, “A comprehensive framework for image inpainting”, Image Processing, IEEE Transactions on, vol. 19, no. 10, **(2010).**

[22] Y. Wei, M. Levoy, “Fast texture synthesis using tree-structured vector quantization”, Proceedings of the 27th annual conference on Computer graphics and interactive techniques. ACM Press /Addison-Wesley Publishing Co., (**2000**), pp. 479-488.

[23] Y. Wei, J. Han, K. Zhou, H. Bao, B. Guo and H. Y. Shum, "Inverse texture synthesis", ACM Transactions on Graphics, ACM, vol. 27, no. 3, **(2008).**
[24] V. Kwatra, A. Schödl, I. Essa, G. Turk and A. Bobick, "Graphcut textures: image and video synthesis using graph cuts", ACM Transactions on Graphics, ACM, vol. 22, no. 3, **(2003).**
[25] V. Kwatra, I. Essa, A. Bobick and N. Kwatra, "Texture optimization for example-based synthesis", ACM Transactions on Graphics, ACM, vol. 24, no. 3, **(2005).**
[26] J. Kopf, C. W. Fu, D. Cohen-Or, O. Deussen, D. Lischinski and T. T. Wong, "Solid texture synthesis from 2d exemplars", ACM Transactions on Graphics, vol. 26, no. 3, **(2007).**
[27] Komodakis, G. Tzirtas, "Image completion using efficient belief propagation via priority scheduling and dynamic pruning", IEEE Transactions on Image Processing, vol. 16, no. 11, **(2007).**
[28] Y. Wexler, E. Shechtman and M. Irani, "Space-time completion of video", Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 29, no. 3, **(2007).**
[29] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman and P. Sen, "Image melding: combining inconsistent images using patch-based synthesis", ACM Transactions on Graphics, vol. 31, no. 4, **(2012).**
[30] M. Ashikhmin, "Synthesizing natural textures", Proceedings of the 2001 Symposium on Interactive 3D Graphics, ACM Press, **(2001)**, New York, USA.
[31] X. Tong, J. Zhang and L. Liu, "Synthesis of bidirectional texture functions on arbitrary surfaces", ACM Transactions on Graphics, Proceedings of ACM SIGGRAPH, vol. 21, no. 3, **(2002).**
[32] K. M. He and J. Sun, "Computing nearest-neighbor fields via propagation-assisted kd-trees", IEEE Conference on Computer Vision and Pattern Recognition, **(2012)** RI, USA.
[33] Barnes, E. Shechtman, A. Finkelstein and D. B. Goldman, "PatchMatch: A randomized correspondence algorithm for structural image editing", ACM Transactions on Graphics, Proceedings of ACM SIGGRAPH, vol. 28, no. 3, **(2009).**
[34] Barnes, E. Shechtman and D. Goldman, "The Generalized PatchMatch correspondence algorithm", European Conference on Computer Vision **(2010)** Springer Berlin Heidelberg.

Authors

Mang Xiao, born in 1982, Ph.D., his main research interests include image processing, virtual reality.

Guangyao Li, born in 1965, Ph. D., professor, Ph. D., supervisor. His main research interests include graphics and image research work, virtual reality.

Yunlan Tan, born in 1972, Ph. D., Her main research interest include virtual reality, visualization in scientific computation.

Qing Jie, born in 1990, Mast, Her main research interest include virtual reality, image processing.

