

Image Inpainting Based on Structural Tensor Edge Intensity Model

Jing Wang Yan-Hong Zhou Hai-Feng Sima Zhan-Qiang Huo Ai-Zhong Mi

College of Computer Science and Technology, Henan Polytechnic University, Jiaozuo 454003, China

Abstract: In the exemplar-based image inpainting approach, there are usually two major problems: the unreasonable calculation of priority and only considering the color features in the patch lookup strategy. In this paper, we propose an image inpainting approach based on the structural tensor edge intensity model. First, we use the progressive scanning inpainting method to avoid the image filling order being affected by the priority function. Then, we use the edge intensity model to build the patches similarity function for correctly identifying the local image structure. Finally, the balance operator is used to restrict the excessive propagation of structural information to ensure the correct structural reconstruction. The experimental results show that the our approach is comparable and even superior to some state-of-the-art inpainting algorithms.

Keywords: Exemplar-based technique, image inpainting, structural tensor, edge intensity model, structure propagation, balance operator.

Citation: J. Wang, Y. H. Zhou, H. F. Sima, Z. Q. Huo, A. Z. Mi. Image inpainting based on structural tensor edge intensity model. *International Journal of Automation and Computing*, vol.18, no.2, pp.256–265, 2021. <http://doi.org/10.1007/s11633-020-1256-x>

1 Introduction

Image restoration (i.e., image de-noising, non-blind de-blurring and image inpainting)^[1] can help recover different kinds of images. Among them, image inpainting refers to using the reliable information remaining in the image to recover the target area (missing or damaged parts). The inpainting technique relies on the color and structural information in the image, ensuring the reconstructed image is visually reasonable. Now that digital images have become a part of our lives, people have more requirements for images, i.e., restoring the images that have been damaged due to improper storage and removing unnecessary parts. This technique also plays an important role in the post-processing of videos and movies, i.e., removing watermarks and recovering damaged vintage films^[2, 3]. Because of the wide application of image inpainting, this technology has attracted the attention of a large number of researchers. It is also an important task in the field of computer vision.

In recent years, image inpainting has made great progress, using methods based on partial differential equations (PDEs), exemplar-based techniques and methods

based on sparse representations. The partial differential equation image inpainting algorithm^[4] is a method based on thermal diffusion, of which the curvature-driven diffusions (CDD) model^[5] and the total variation (TV) model^[6] are two typical ones. Li et al.^[7] proposed an improved TV algorithm in which the calculation of the diffusion coefficient is determined by the distance and direction between the pixels in the target area and the pixels in the domain. However, diffusion-based methods are more suitable for inpainting non-textured areas, because the texture information will be blurred as the information diffuses. Yang et al.^[8] applied a structural tensor to the PDE algorithm to improve this defect. The advantage of the algorithm based on partial differential equations^[4] is that it deals with small damaged areas (i.e., scratches). For areas with large damage or images with rich structural information, the algorithm often does not get the expected results. Therefore, an exemplar-based image inpainting method is proposed. In the exemplar-based image inpainting algorithm, determining the repair order of image edges and the patch lookup strategy are general steps. This inpainting method is good at dealing with images with large area damage. But, the exemplar-based image inpainting algorithm has two major problems: the failure of the priority function and the incomplete matching patch search strategy. Many inpainting algorithms for these problems have been proposed and the details of these algorithms have been introduced in [9]. Li et al.^[10] subtracted a constant from the confidence term to avoid the priority function failing. Wang et al.^[11] im-

Research Article

Manuscript received June 13, 2020; accepted September 16, 2020; published online November 28, 2020

Recommended by Associate Editor De Xu

Colored figures are available in the online version at <https://link.springer.com/journal/11633>

© Institute of Automation, Chinese Academy of Sciences and Springer-Verlag GmbH Germany, part of Springer Nature 2020

proved the robustness of the algorithm by introducing regularization factors to adjust the priority function, and using a method which combines a modified sum of squared differences (SSD) and normalized cross-correlation (NCC) to adjust the search strategy for the matching patch. Liu et al.^[12] used the structural tensor to construct a local measurement function to optimize the priority and proposed a matching patch search strategy. The method in [12] contributes to the structural fidelity of the image inpainting results. In [13], the size of the matching patch is calculated by using the structure tensor, which is more suitable for the inpainting of various texture structure images. Zhang et al.^[14] proposed a nonlocal patch tensor-based visual data completion algorithm, and the experimental results on real-world datasets showed the superiority of the method. In addition, tensor theory is widely used. It was used in object recognition in 1988^[15]. The semi-tensor product of matrices is used to show whether a hyper-networked evolutionary games (HNEG) is potential and how to calculate the potential^[16]. In order to consider the overall structure contour of the image, some of the literature^[17] used image pyramid and Patch-Match algorithms^[18] to solve the problem of correct reconstruction of texture and structure in image. Liu et al.^[19] combined the image pyramid with the Criminisi algorithm^[20] in order to use the inpainting of the upper layer of the pyramid to guide the inpainting of the lower layer. This method uses the results of the upper layer to improve the priority function to obtain a more reliable inpainting order. In addition, research found that the same sparsity exists in the target area and the known area of the image. So, there is also an inpainting method based on sparse representation^[21, 22]. In recent years, deep learning is becoming popular. It is widely applied in many domains and has obtained many encouraging results. Yao et al.^[23] introduced the application of deep learning in healthcare extensively. Ha et al.^[1] gave an overview of recent advances in deep learning-based models and methods that have been applied to single image super-resolution tasks. Wang et al.^[24] proposed an image inpainting method for large-scale irregular masks.

In exemplar-based image inpainting algorithms, the quality of the inpainting is affected by both the edge filling priority and the patch lookup strategy. While the filling progresses, both the data term and the confidence term will gradually approach zero, causing the priority function to fail. Moreover, the patch lookup strategy relies on the single color feature to search for the patch. For images with complex texture and structure, it is easy to match the wrong patches in inpainting. Therefore, for the two problems mentioned above, no matter which of these two parts has a slight error, it will directly affect the subsequent filling effect. The inpainting of this algorithm depends on the previously filled information. When an error occurs in the previous inpainting, the subsequent inpainting information also changes. It will cause

a chain reaction in the later inpainting, which leads to inconsistent results. In addition, this algorithm is also susceptible to the shape of the target area. When the selected target area changes a little bit, the pixels at the edges will change greatly. Therefore, this paper proposes an image inpainting technique based on the structural tensor edge intensity model. It uses a progressive scan inpainting method to avoid calculating the priority of edge filling patches. When it finds the target area, it starts to search for matching patches. In order to accurately identify the structural information in the image, the constructed edge strength model is used to search for matching patches to ensure patches are found accurately. Framework of the proposed algorithm is shown in Fig. 1.

2 Related work

2.1 Symbol definition

As shown in Fig. 2, Ω is the source area, Φ is the target area, Ψ_x (a square patch centered on the pixel x , the size is 9×9 pixels) is the scan patch, and Ψ_p (the size is 6×6 pixels) is the patch to be filled. The area Ψ_z is a source area adjacent to the patch Ψ_p . As shown in Fig. 2(a), the area Ψ_z is divided into two parts: A (the blue box on the left of patch Ψ_p) and B (the green box above patch Ψ_p). When the scan patch Ψ_x is in the state of Fig. 2(b), the patch Ψ_x is the patch to be matched, the patch Ψ_y is the best matching patch, and the patch Ψ_q is the best filling patch. In Fig. 2(c), h is the length of the entire target area Φ , w is the width of the entire target area Φ , \hat{h} is the distance from the bottom of the patch Ψ_x to the lower boundary of the target area Φ , and \hat{w} is the distance from the right side of the patch Ψ_x to the right boundary of the target area Φ . Let \hat{h}/h be the value L_h and \hat{w}/w be the value L_w . The adjacent square patches to the left and above of the patch to be filled Ψ_p are denoted as Ψ_p^A and Ψ_p^B (the size is 6×6 pixels).

2.2 Edge intensity model

To define the structure tensor, let the gradient of the image be

$$\nabla I = [I_x, I_y]^T \quad (1)$$

where I_x and I_y are partial deviations of pixels in the x direction and y direction. The structural tensor E of the image can be defined as

$$E = \nabla I \nabla I^T = \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}. \quad (2)$$

E is a symmetric and semi-positive two-dimensional matrix, it has two non-negative eigenvalues λ_1 and λ_2 .

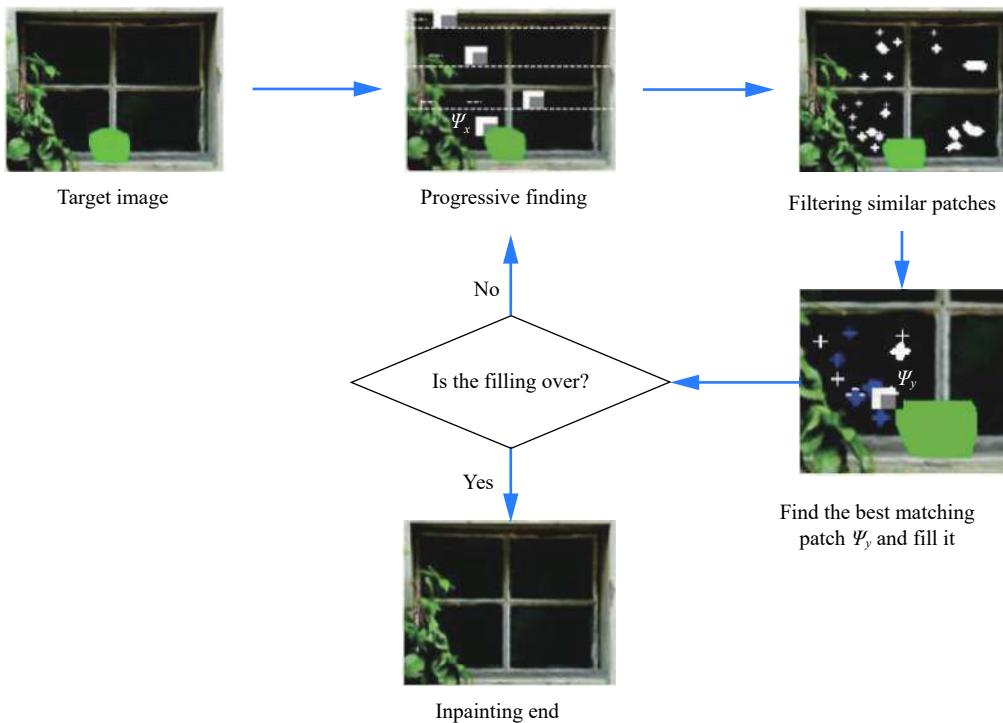
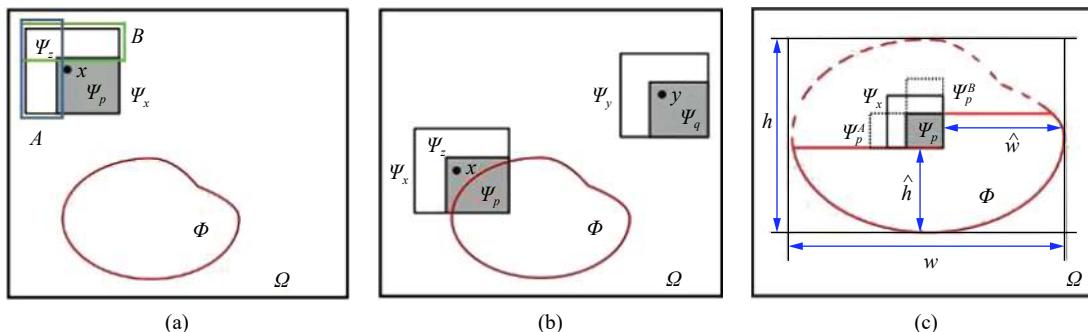


Fig. 1 Framework of the proposed algorithm

Fig. 2 Symbol definition: (a) Source area Ω , target area Φ and scan patch Ψ_x ; (b) Patch to be matched Ψ_x and the best matching patch Ψ_y ; (c) Schematic diagram of the degree of image to be inpainted.

The edge intensity is using λ_1 and λ_2 to define the coherence of the image data. The edge intensity at pixel (i, j) is

$$H(i, j) = (\lambda_1 - \lambda_2)^2. \quad (3)$$

The eigenvalues λ_1 and λ_2 can be used to analyze the local image structure. When both eigenvalues are approaching zero, it means that the gray value has little changes in all directions and they are located in a flat area. When one eigenvalue is much larger than the other eigenvalue (toward zero), it means that the image has a significant edge structure. When both eigenvalues are greater than zero and one eigenvalue is much larger than the other eigenvalue, it means that a corner area exists.

In the traditional exemplar-based image inpainting algorithm, only the color features are used to search for matching patches. As a result, the structural information around the target area cannot be correctly identified. As shown in Fig. 3, the source area of the patch to be

matched Ψ_x already has a straight line structure and a part of the corner structure. If only a single color feature is used to find the best matching patch, it is easy to miss the corner information near the patch Ψ_x to be matched. So, the wrong matching patch Ψ_{y2} is found, and the correct matching patch Ψ_{y1} is missed. As a result, the straight line structure is inpainted, but the information of the corner points is lost. This causes the image inpainting to fail.

The structure tensor can identify the characteristics of different texture structures in the image. By calculating the edge intensities of the patches Ψ_x and Ψ_y (represented by the edge intensities at the central pixels $x(i, j)$ and $y(i, j)$ of patches Ψ_x and Ψ_y), the accuracy of identifying the structural information in the target image is improved. The two edge intensities are denoted as $H(i_x, j_x)$ and $H(i_y, j_y)$. In order to judge the coherence between the patch to be matched Ψ_x and the matching patch Ψ_y , an edge intensity model of the image is constructed:

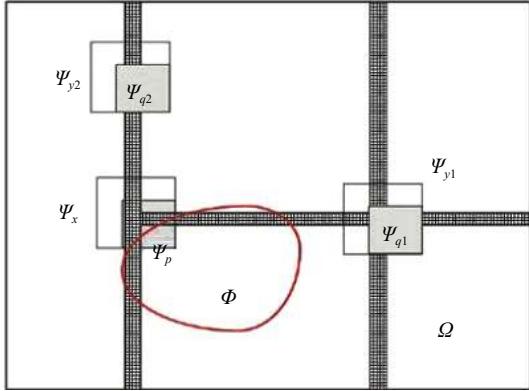


Fig. 3 Incorrect reconstruction of patch Ψ_x local structure

$$\bar{H}(x, y) = \frac{(H(i_x, j_x) + H(i_y, j_y))^2}{\omega} \quad (4)$$

where ω is a constant ($\omega = 0.001$). From (3) and the characteristics of the structural tensor eigenvalues, it can be seen that when the value of $\bar{H}(x, y)$ tends to zero or equals zero, it means that Ψ_x and Ψ_y have low coherence and the pixels are located in the flat region. If the value of $\bar{H}(x, y)$ is much larger than zero, it means that Ψ_x and Ψ_y are highly coherent and there are structure information around pixels.

3 Algorithm

There are usually two major problems for the exemplar-based image inpainting approach: the priority function is prone to failure and the patch lookup strategy is not perfect. This paper proposes an image inpainting technique based on the structural tensor edge intensity model method. The implementation steps are as follows.

3.1 Progressive scanning

In exemplar-based image inpainting algorithms, the priority is determined by the product of the data term and the confidence term. However, as the filling progresses, both the data term and the confidence term will gradually approach zero, making the priority function invalid. As a result, the order of edge filling cannot be determined, and it can cause filling errors easily.

In order to avoid being affected by the priority function, the algorithm uses a progressive scan method in inpainting. The algorithm scans the image progressively with scan patch Ψ_x , and moves one pixel length at a time. When the scan patch Ψ_x finds the target area Φ , it continues to move the scan patch Ψ_x until Ψ_x and Φ have the largest coincident area.

When it satisfies (5), then it starts searching for matching patches.

$$\Psi_z \cap \Phi = \emptyset. \quad (5)$$

3.2 Accurately identify local structural information

The single color feature is considered in the traditional patch lookup strategy. It ignores the structural information of the image. In this paper, the edge intensity model is introduced to identify the local image structural features. The patches similarity function is defined as

$$\bar{D}(\Psi_x, \Psi_y) = D(\Psi_x, \Psi_y) + \alpha \times \bar{H}(x, y) \quad (6)$$

where α is the intensity adjustment factor (empirical value $\alpha=10$), $\bar{H}(x, y)$ is the edge intensity between the patch to be matched Ψ_x and the matching patch Ψ_y , $D(\Psi_x, \Psi_y)$ is expressed as

$$D(\Psi_x, \Psi_y) = \sum \sqrt{(R_{\Psi_x} - R_{\Psi_y})^2 + (G_{\Psi_x} - G_{\Psi_y})^2 + (B_{\Psi_x} - B_{\Psi_y})^2}. \quad (7)$$

The SSD distance $D(\Psi_x, \Psi_y)$ is the similarity between each patch Ψ_y in the source area and patch Ψ_x , where R , G and B represent the pixel values corresponding to each channel.

In this paper, the scan patch Ψ_x also provides the basis for the calculation of the edge intensity $H(i_x, j_x)$. In Fig. 2(b), the area Ψ_z contains more reliable information, which can accurately identify the texture and structure around the patch to be filled Ψ_p , then search for the best matching patch Ψ_y according to (6) all over the image. In order to improve the accuracy of matching patches, a filtering strategy is defined as

$$\Psi_y < \arg \min_{\Psi_y \in \Omega} \bar{D}(\Psi_x, \Psi_y) \times err \quad (8)$$

where $\bar{D}(\Psi_x, \Psi_y)$ is the similarity between the patch to be matched Ψ_x and the matching patch Ψ_y , err is the error adjustment factor: $err = \lambda + \beta \times H(i_x, j_x)$, experience values $\lambda = 1.3$, $\beta = 10$, $H(i_x, j_x)$ is the edge intensity of the patch to be matched Ψ_x .

After filtering, there may be three cases of the matching patch Ψ_y that exist. The area Ψ_z is divided into two areas A and B (as shown in Fig. 2(a)).

1) The matching patch Ψ_y matches the image information of the corresponding area A and area B , then it is the best matching patch with the best filling performance.

2) The matching patch Ψ_y matches the image information of the corresponding area A or area B , then some of them have reliable filling information.

3) The matching patch Ψ_y matches part of the image information of the corresponding area A and area B , but not all of them completely match.

If the filtered matching patch Ψ_y does not exist in 1), the best matching patch needs further matching from 2)

and 3). Through analysis, if there is a matching difference between the two matching patches, there will be structural information in one of the two areas. This indicates that the image information of areas A and B (as shown in Fig. 2(a)) should have a large difference. Structure information is the skeleton of the image information, so a direction with structure information is chosen in these two areas to inpainting first.

3.3 Balance factor

If structural information is allowed to spread excessively, the inpainting will fail. So we define a balance factor to limit the excessive expansion of the structure. This balance factor is based on the image gradient and the degree of image inpainting. First determine the degree of inpainting of the current target area Φ . The degree of inpainting is defined as the ratio of the width of the remaining inpainting area to the width of the whole target area. The smaller the ratio is, the greater the degree of inpainting in a certain direction is. The higher the degree of repair, the less difficult it is for inpainting. Therefore, the direction with a large degree of inpainting is inpainted first. As shown in Fig. 2(c), calculate the inpainting degree of the patch to be matched Ψ_x in both vertical and horizontal directions. Let L_h and L_w denote the inpainting degree of the patch to be matched Ψ_x in these two directions. Note the modulus values of the gradients of Ψ_p^A and Ψ_p^B (adjacent patches of the patch Ψ_p to be filled, as shown in Fig. 2(c)) are $|\mathbf{I}|_A$ and $|\mathbf{I}|_B$. The larger the modulus value is, the stronger the structural information contained in the image is. Therefore, the area with strong structural information is preferentially inpainted.

The balance factor determines the direction in which the patch to be matched Ψ_x is preferentially matched. As shown in Fig. 2(a), it performs priority matching according to the matching information in the area A or the area B . The balance factor V can be expressed as

$$V = \frac{|\mathbf{I}|}{L} \quad (9)$$

where $|\mathbf{I}|$ is $|\mathbf{I}|_A$ or $|\mathbf{I}|_B$, and L is L_h or L_w .

Finally, use (7) to calculate the SSD distance $D(\Psi_p^r, \Psi_q)$ of Ψ_p^A or Ψ_p^B (as shown in Fig. 2(c)) and the filled patch Ψ_q (r is A or B). Take the average of these distances as Ave :

$$Ave = \frac{\sum_{i=1}^n D(\Psi_p^r, \Psi_q)}{n} \quad (10)$$

where Ψ_p^r is Ψ_p^A or Ψ_p^B , n is the total number of matching patches in Ψ_y (matching patches filtered by (8)). The larger the SSD distance between the two patches is, the smaller the similarity between the two patches is. The smaller the SSD distance is, the higher the similarity

between the two patches is. Therefore, among the candidate patches whose distance $D(\Psi_p^r, \Psi_q)$ is smaller than Ave , we select the matching patch which is most similar to the patch Ψ_p as the best filling patch Ψ_q , and the pixels in the patch Ψ_q are copied to the corresponding position in the patch Ψ_p .

The pseudo code of the algorithm in this paper is shown in Algorithm 1. Fig. 4 presents the process of four groups searching for matching patch in the image inpainting of Sill. In Fig. 4(a) State 1, the white patches represent the matching patches left after being filtered by (8). In Fig. 4(b) State 2, the blue patches represent further filtering of white matching patches using the balance factor V . In Fig. 4(c) State 3, the red patch is the best matching patch. It can be seen from the four pairs of matching patches search process diagrams that the matching patches search is correct whether it is a flat area or a structural area. It shows that our search strategy is effective in finding the best matching patch. The complete image inpainting process of Sill is shown in the animation (Online Resource 1).

Algorithm 1. Image inpainting based on structural tensor edge intensity model

Input: Target image, marker color

Output: Inpainted image

1) Use the 9×9 scan patch Ψ_x as the basic patch for scanning. Scan the target image progressively from top to bottom and from left to right, when the scan patch Ψ_x finds the target area Φ , it continues to move the scan patch Ψ_x ;

2) **while** $\Psi_z \cap \Phi = \emptyset$ **do**

i) Use patch similarity (6) to perform a global search on the image and search for matching patches for the patch to be matched Ψ_x ;

ii) Use (8) to filter the matching patches in Step i);

iii) Determine the priority matching direction by calculating the balance factor V of the patch to be matched Ψ_x ;

iv) Use (7) to calculation the SSD distance $D(\Psi_p^r, \Psi_q)$ between the patch Ψ_p^A or Ψ_p^B and the filling patch Ψ_q , and average these distances using (10);

v) Among the matching patches whose distance $D(\Psi_p^r, \Psi_q)$ is smaller than the average Ave , the matching patch most similar to the patch to be filled Ψ_p is the best filling patch Ψ_q . Then, the pixels in the best filling patch Ψ_q are correspondingly copied to the corresponding positions in the patch to be filled Ψ_p .

3) **end while**

4) Repeat the above steps until the image is inpainted.

4 Experimental results

Visual perception is generally used to evaluate the results of image inpainting. In addition, it can be illustrated by using the structural similarity index measure (SSIM),



Fig. 4 Process of searching for matching patch in the image inpainting of Sill: (a) State 1; (b) State 2; (c) State 3.

the peak signal to noise ratio (PSNR) value and the time required for inpainting. The algorithm demonstrates the effectiveness of the experiments in this paper by comparing with the methods in [20], [11], and [10].

To test our method, we check its inpainting ability in various scene images as shown Figs. 5–8. Among them, the column (a) is the target image, the columns (b), (c) and (d) are the inpainting results of the methods in [20], [11] and [10], and the column (e) is the inpainting results of our method.

4.1 Damaged image inpainting

Figs. 5 and 6 show the inpainting of damaged images. They are the fence, window, stair, island, mountain and highway images, respectively.

In Fig. 5, there are some images with complex texture and structures. In Figs. 5(b)–5(d), we can see that the results of methods in [20], [11] and [10] show incorrect information. This is because failure to consider the surrounding features of the image structure leads to repair failure. For example, in the inpainting of the images fence and window, the methods of [20] and [11] have broken the linear structure and cross structure. The method of [10]

cannot recover the linear structure of fence and stair, but it reconstructs the cross structure of window. In the inpainting of stair, the methods in [20] and [10] are incorrect in rebuilding the linear structure of the step. However, our method restores the incorrect structure of the images of fence, window and stair.

There are three scene images in Fig. 6. In the inpainting of images island and mountain, the method in [20] copies the wrong information into the sky, and the methods in [11] and [10] are wrong in rebuilding trees and mountains. In this case, our method can rebuild the trees and sky. And our method and methods in [20], [11] and [10] have achieved good results in the inpainting of highway.

Tables 1 and 2 are the experimental evaluation of the above six groups using the indicators PSNR and SSIM. These two indicators can reflect the inpainting effect of different methods. The results show that our method is superior to the others in [20], [11] and [10].

4.2 Object removal

The inpainting images of the removed object are shown in Figs. 7 and 8. These are images inpainted with

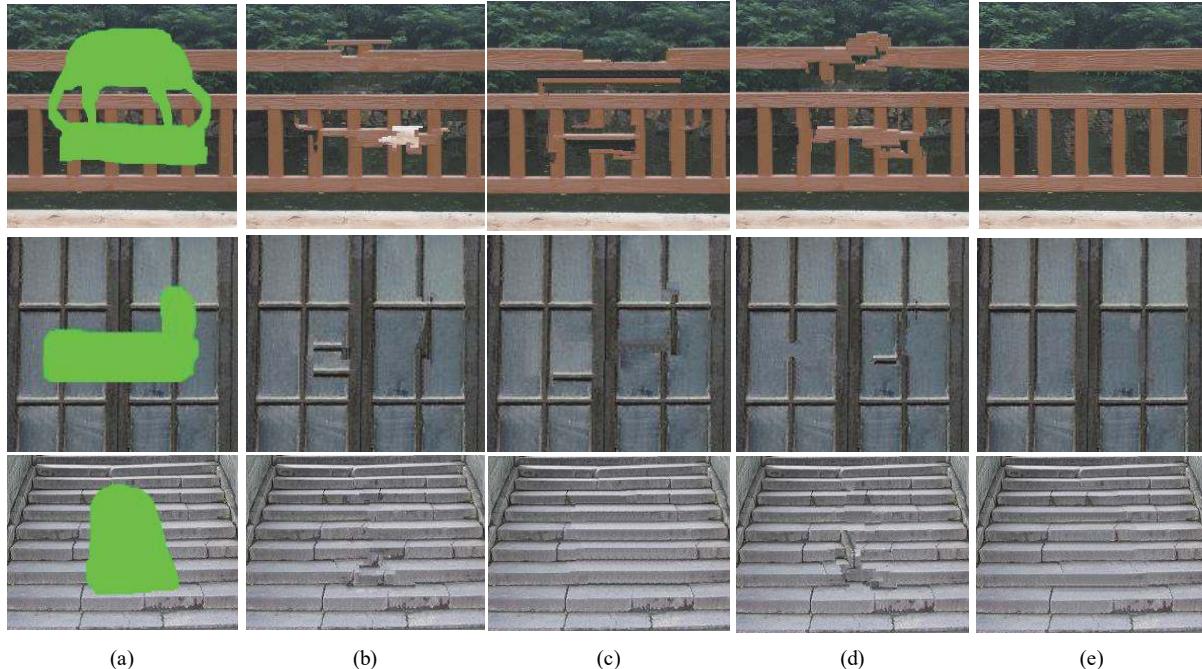


Fig. 5 Inpainting results of fence, window and stair: (a) Target image; (b) Method in [20]; (c) Method in [11]; (d) Method in [10]; (e) Our method.

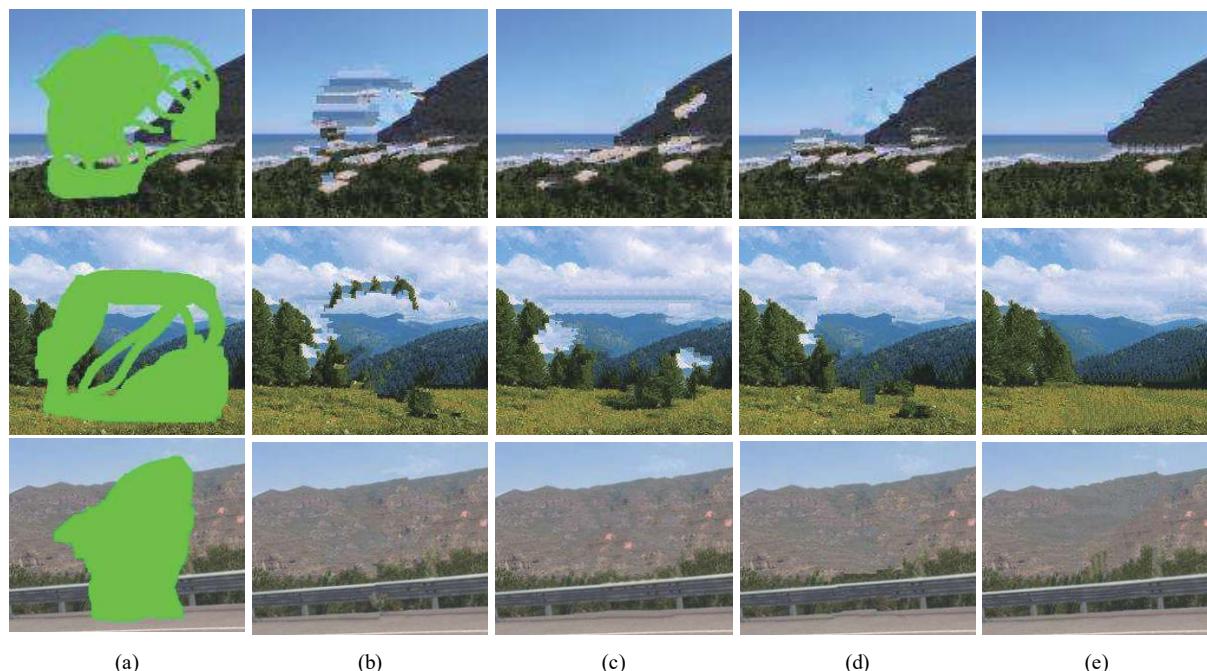


Fig. 6 Inpainting results of island, mountain and highway: (a) Target image; (b) Method in [20]; (c) Method in [11]; (d) Method in [10]; (e) Our method.

highly textured irregular patterns. They are the girl, woman, fox, people, tourist and boat images, respectively. The object removal means to remove the extra objects in the image, i.e., the people in the images girl and woman, the fox lying on wood, the people and car in the images people and tourist, and the sailing ship in the image boat.

Table 3 lists a time comparison of several methods. From Table 3, we can see that compared with the three

methods in [20], [11] and [10], our method can not only save inpainting time but also ensure that inpainting results meet the visual requirements.

5 Conclusions

This paper proposes an image inpainting algorithm based on the structure tensor edge intensity model. It

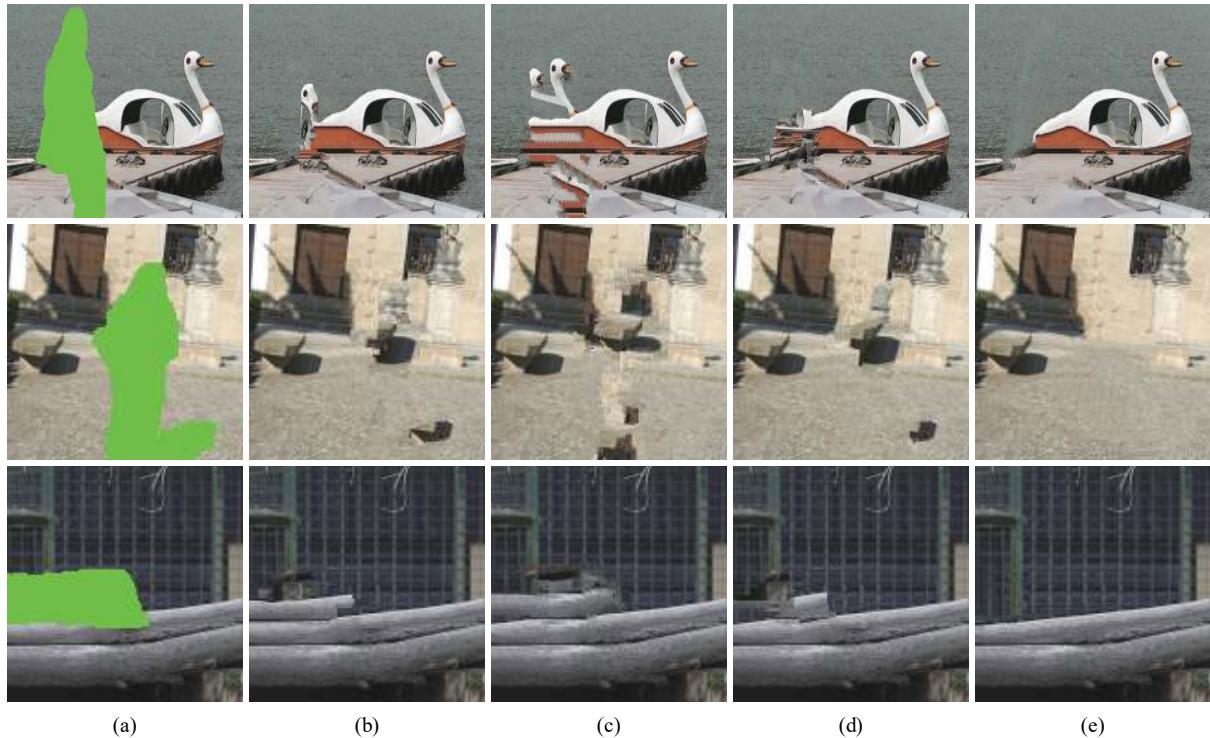


Fig. 7 Object removal for images girl, woman and fox: (a) Target image; (b) Method in [20]; (c) Method in [11]; (d) Method in [10]; (e) Our method.



Fig. 8 Object removal for images people, tourist and boat: (a) Target image; (b) Method in [20]; (c) Method in [11]; (d) Method in [10]; (e) Our method.

uses a progressive scan method to avoid the edge filling order being affected by the priority function. An edge intensity model is introduced to the patch lookup strategy. Meanwhile, the structural information and color informa-

tion of the image are considered to ensure the correctness of the matching patch. The effectiveness of the algorithm is proved by multiple simulation experiments in Matlab. Our experiments are carried out on an Intel core

Table 1 PSNR of different algorithms to inpainting images (dB)

Image (size)	Method in [20]	Method in [11]	Method in [10]	Our method
Fence (227×227)	26.456 7	27.423 8	27.966 8	33.687 1
Window (202×256)	32.422 3	31.853 2	34.822 0	38.182 0
Stair (392×258)	30.877 8	32.319 4	29.982 3	32.906 1
Island (128×128)	23.355 4	25.693 0	26.408 0	30.105 5
Mountain (300×200)	22.167 7	21.977 8	25.621 8	27.969 6
Highway (240×212)	30.855 3	30.699 0	29.894 5	31.689 8

Table 2 SSIM of different algorithms to inpainting images

Image (size)	Method in [20]	Method in [11]	Method in [10]	Our method
Fence (227×227)	0.884 3	0.842 6	0.873 9	0.902 6
Window (202×256)	0.920 1	0.905 9	0.931 9	0.946 8
Stair (392×258)	0.913 9	0.922 1	0.906 6	0.928 0
Island (128×128)	0.844 4	0.856 8	0.878 8	0.887 9
Mountain (300×200)	0.810 2	0.787 5	0.854 7	0.859 2
Highway (240×212)	0.844 6	0.840 1	0.829 0	0.854 5

Table 3 Image restoration time by different methods (s)

Image (size)	Method in [20]	Method in [11]	Method in [10]	Our method
Fence (227×227)	41.123 0	695.431 9	43.568 3	40.568 5
Window (202×256)	12.357 8	139.361 6	13.039 8	12.273 6
Stair (392×258)	15.130 6	272.137 5	17.850 0	14.194 3
Island (128×128)	15.299 3	112.654 4	15.844 1	14.047 6
Mountain (300×200)	38.969 8	540.881 1	50.818 5	36.071 0
Highway (240×212)	35.190 1	758.211 8	45.167 3	34.008 5
Girl (384×256)	54.508 4	1 861.6	66.005 5	43.523 1
Woman (128×128)	8.621 2	71.90.85	9.278 3	8.321 8
Fox (128×128)	4.841 1	42.825 7	4.798 0	4.705 7
People (128×128)	4.840 5	34.448 8	4.911 8	4.685 4
Tourist (240×180)	21.744 0	371.810 6	22.071 1	20.694 6
Boat (128×128)	10.838 3	62.356 9	11.717 5	10.185 4

i5CPU@1.6GHz. For the reconstruction of damaged images and the removal of objects, the results of inpainting are visually reasonable. In addition, the PSNR and SSIM indicators are used to compare with different algorithms. Experiments show that the algorithm we proposed in this paper is satisfactory for image inpainting of different types of structures. Moreover, by comparing the inpainting time of several algorithms, it also illustrates the high efficiency of our algorithm.

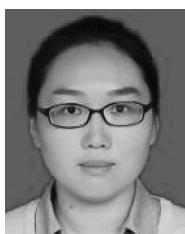
Acknowledgements

This work was supported by National Science Foundation of China (Nos. 61401150, 61602157 and 61872311), Key Science and Technology Program of Henan Province (Nos. 182102210053 and 202102210167), Excellent Young Teachers Program of Henan Polytechnic University (No. 2019XQG-02).

References

- V. K. Ha, J. C. Ren, X. Y. Xu, S. Zhao, G. Xie, V. Masero, A. Hussain. Deep learning based single image super-resolution: A survey. *International Journal of Automation and Computing*, vol. 16, no. 4, pp. 413–426, 2019. DOI: [10.1007/s11633-019-1183-x](https://doi.org/10.1007/s11633-019-1183-x).
- M. Wang, R. C. Hong, X. T. Yuan, S. C. Yan, T. S. Chua. Movie2Comics: Towards a lively video content presentation. *IEEE Transactions on Multimedia*, vol. 14, no. 3, pp. 858–870, 2012. DOI: [10.1109/TMM.2012.2187181](https://doi.org/10.1109/TMM.2012.2187181).
- J. B. Huang, S. B. Kang, N. Ahuja, J. Kopf. Temporally coherent completion of dynamic video. *ACM Transactions on Graphics*, vol. 35, no. 6, Article number 196, 2016. DOI: [10.1145/2980179.2982398](https://doi.org/10.1145/2980179.2982398).
- M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester. Image inpainting. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, ACM, New York, USA, pp. 417–424, 2000. DOI: [10.1145/344779.344972](https://doi.org/10.1145/344779.344972).
- T. F. Chan and J. H. Shen. Nontexture inpainting by curvature-driven diffusions. *Journal of Visual Communication and Image Representation*, vol. 12, no. 4, pp. 436–449, 2001. DOI: [10.1006/jvc.2001.0487](https://doi.org/10.1006/jvc.2001.0487).
- J. H. Shen and T. F. Chan. Mathematical models for local nontexture inpaintings. *SIAM Journal on Applied Mathematics*, vol. 62, no. 3, pp. 1019–1043, 2002. DOI: [10.1137/s0036139900368844](https://doi.org/10.1137/s0036139900368844).
- K. S. Li, Y. S. Wei, Z. Yang, W. H. Wei. Image inpainting algorithm based on TV model and evolutionary algorithm. *Soft Computing*, vol. 20, no. 3, pp. 885–893, 2016. DOI: [10.1007/s00500-014-1547-7](https://doi.org/10.1007/s00500-014-1547-7).
- X. H. Yang, B. L. Guo, Z. L. Xiao, W. Liang. Improved structure tensor for fine-grained texture inpainting. *Signal Processing: Image Communication*, vol. 73, pp. 84–95, 2019. DOI: [10.1016/j.image.2018.02.006](https://doi.org/10.1016/j.image.2018.02.006).
- P. Buyssens, M. Daisy, D. Tschumperlé, O. Lézoray. Exemplar-based inpainting: Technical review and new heuristics for better geometric reconstructions. *IEEE Transactions on Image Processing*, vol. 24, no. 6, pp. 1809–1824, 2015. DOI: [10.1109/TIP.2015.2411437](https://doi.org/10.1109/TIP.2015.2411437).
- X. F. Li, J. Wang, H. M. Liu, Z. H. Wang. Image inpainting using feature precedence and patch matching. *Journal of Computer-Aided Design & Computer Graphics*, vol. 28, no. 7, pp. 1131–1137, 2016. DOI: [10.3969/j.issn.1003-9775.2016.07.012](https://doi.org/10.3969/j.issn.1003-9775.2016.07.012). (in Chinese)
- J. Wang, K. Lu, D. R. Pan, N. He, B. K. Bao. Robust object removal with an exemplar-based image inpainting approach. *Neurocomputing*, vol. 123, pp. 150–155, 2014. DOI: [10.1016/j.neucom.2013.06.022](https://doi.org/10.1016/j.neucom.2013.06.022).
- Y. Liu, C. J. Liu, H. L. Zou, S. S. Zhou, Q. Shen, T. T. Chen. A novel exemplar-based image inpainting algorithm. In *Proceedings of International Conference on Intelligent Networking and Collaborative Systems*, IEEE, Taipei, China, pp. 86–90, 2015. DOI: [10.1109/INCoS.2015.15](https://doi.org/10.1109/INCoS.2015.15).

- [13] X. H. Bi, H. M. Liu, G. M. Lu, J. Wei, Y. Chao. Exemplar-based image inpainting using automatic patch optimization. In *Proceedings of the 2nd International Conference on Video and Image Processing*, ACM, Hong Kong, China, pp. 128–133, 2018. DOI: [10.1145/3301506.3301549](https://doi.org/10.1145/3301506.3301549).
- [14] L. F. Zhang, L. C. Song, B. Du, Y. P. Zhang. Nonlocal low-rank tensor completion for visual data. *IEEE Transactions on Cybernetics*, published online. DOI: [10.1109/TCYB.2019.2910151](https://doi.org/10.1109/TCYB.2019.2910151).
- [15] M. El Mallahi, A. Zouhri, A. El Affar, A. Tahiri, H. Qjidaa. Radial Hahn moment invariants for 2D and 3D image recognition. *International Journal of Automation and Computing*, vol. 15, no. 3, pp. 277–289, 2018. DOI: [10.1007/s11633-017-1071-1](https://doi.org/10.1007/s11633-017-1071-1).
- [16] T. Liu, Y. H. Wang, D. Z. Cheng. Dynamics and stability of potential hyper-networked evolutionary games. *International Journal of Automation and Computing*, vol. 14, no. 2, pp. 229–238, 2017. DOI: [10.1007/s11633-017-1056-0](https://doi.org/10.1007/s11633-017-1056-0).
- [17] A. Newson, A. Almansa, Y. Gousseau, P. Pérez. Non-local patch-based image inpainting. *Image Processing on Line*, vol. 7, pp. 373–385, 2017. DOI: [10.5201/ipol.2017.189](https://doi.org/10.5201/ipol.2017.189).
- [18] C. Barnes, E. Shechtman, A. Finkelstein, D. B. Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics*, vol. 28, no. 3, Article number 24, 2009. DOI: [10.1145/1531326.1531330](https://doi.org/10.1145/1531326.1531330).
- [19] H. M. Liu, X. H. Bi, G. M. Lu, W. L. Wang, Z. Y. Zhang. Screen window propagating for image inpainting. *IEEE Access*, vol. 6, pp. 61761–61772, 2018. DOI: [10.1109/ACCESS.2018.2876161](https://doi.org/10.1109/ACCESS.2018.2876161).
- [20] Criminisi A, Pérez P, Toyama K. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing*, vol. 13, no. 9, pp. 1200–1212, 2004. DOI: [10.1109/TIP.2004.833105](https://doi.org/10.1109/TIP.2004.833105).
- [21] Y. Xu, L. C. Yu, H. T. Xu, H. Zhang, T. Nguyen. Vector sparse representation of color image using quaternion matrix analysis. *IEEE Transactions on Image Processing*, vol. 24, no. 4, pp. 1315–1329, 2015. DOI: [10.1109/TIP.2015.2397314](https://doi.org/10.1109/TIP.2015.2397314).
- [22] V. Naumova, K. Schnass. Dictionary learning from incomplete data for efficient image restoration. In *Proceedings of the 25th European Signal Processing Conference*, IEEE, Kos, Greece, pp. 1425–1429, 2017. DOI: [10.23919/EUSIPCO.2017.8081444](https://doi.org/10.23919/EUSIPCO.2017.8081444).
- [23] Z. J. Yao, J. Bi, Y. X. Chen. Applying deep learning to individual and community health monitoring data: A survey. *International Journal of Automation and Computing*, vol. 15, no. 6, pp. 643–655, 2018. DOI: [10.1007/s11633-018-1136-9](https://doi.org/10.1007/s11633-018-1136-9).
- [24] N. Wang, S. H. Ma, J. Y. Li, Y. P. Zhang, L. F. Zhang. Multistage attention network for image inpainting. *Pattern Recognition*, vol. 106, Article number 107448, 2020. DOI: [10.1016/j.patcog.2020.107448](https://doi.org/10.1016/j.patcog.2020.107448).



Jing Wang received the B.Sc. degree in computer science and technology from Henan University of Science and Technology, China in 2006, and the Ph.D. degree in computer application technology from College of Computing and Communication Engineering, Graduate University of Chinese Academy of Science, China in 2012. Currently, she is an associate pro-

fessor in College of Computer Science and Technology, Henan Polytechnic University, China.

Her research interests include image processing computer vision and machine learning.

E-mail: wjasmine@hpu.edu.cn
ORCID iD: 0000-0002-3288-2111



Yan-Hong Zhou received the B.Eng. degree in computer science and technology from Fuyang Teachers College, China in 2018. Currently, she is a master student in software engineering at College of Computer Science and Technology, Henan Polytechnic University, China.

Her research interests include image processing and computer vision.

E-mail: zhouchanhong_zyh123@163.com



Hai-Feng Sima received the B.Eng. and M.Eng. degrees in computer science from Zhengzhou University, China in 2004 and 2007, respectively, and the Ph.D. degree in software and theory from Beijing Institute of Technology, China in 2015. Since 2007, he has been with Faculty of Henan Polytechnic University, China, and is currently a lecturer with College of Computer Science and Technology, Henan Polytechnic University, China.

His current research interests include pattern recognition, image processing, image segmentation and image classification.

E-mail: smhf@hpu.edu.cn (Corresponding author)
ORCID iD: 0000-0002-2049-3637



Zhan-Qiang Huo received the B.Sc. degree in mathematics and applied mathematics from the Hebei Normal University of Science and Technology, China in 2003. He received the M.Sc. degree in computer software and theory and the Ph.D. degree in circuit and system from Yanshan University, China in 2006 and 2009. Currently, he is an associate professor in the College of Computer Science and Technology, Henan Polytechnic University, China. His research interests include computer vision and machine learning.

E-mail: hzq@hpu.edu.cn



Ai-Zhong Mi received the M.Sc. degree in computer and application from Guangxi University, China in 2005, and the Ph.D. degree in computer application technology from University of Science and Technology Beijing, China in 2009. He is currently an associate professor in College of Computer Science and Technology, Henan Polytechnic University, China.

His research interests include pattern recognition and network security.

E-mail: miaizhong@163.com