A fast image inpainting algorithm based on an adaptive scanning strategy

H. R. Guo[1,*], W. H. Wang[1]

[1]School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, Henan 454000, P R China

Abstract

OBJECTIVES: In exemplar-based image inpainting algorithms, there are often issues with the calculation of patch similarity for matching, suboptimal strategies for selecting matching patches, and low inpainting speed.
METHODS: First, large-scale scanning patches are used to scan the known areas of the image line by line to solve the problem of slow scanning speed. When the scanning reaches the area to be repaired, the large-scale scanning patches are adaptively changed to small-scale scanning patches to continue scanning, improving the accuracy of local information inpainting in the image. Then, an improved weighted similarity calculation formula is used to search for similar matching patches, and the search range gradually expands from local to global to improve computational efficiency. To further improve the correctness of the matching patch selection, multiple matching criteria and filtering strategies are used for selection. Finally, the direction of priority inpainting is determined by the stability factor, followed by filling.
RESULTS: The experimental results show that the inpainting effect of the proposed method is significantly improved in subjective vision, and the structural similarity (SSIM), peak signal-to-noise ratio (PSNR), and inpainting speed of the inpainting results are all improved.
CONCLUSION: For damaged images, texture images, and target removal images, the proposed method has a better inpainting effect and higher inpainting speed than the other three advanced methods.

1. Introduction

As technology for people to record information, the image has gradually become a significant part of people's life. In terms of conveying information, images are the most intuitive, but the prerequisite for the intuitiveness of images is completeness [1]. During the process of image generation, storage, compression, and transmission, it may be damaged for a variety of reasons, which will make the image lack integrity, and then reduce the value of the image, which requires image processing technology to inpainting it [2]. At present, image inpainting technology has been widely used in many aspects of society, such as image coding, image enhancement, image compression, digital cultural heritage protection [3, 4], target removal [5, 6], subtitle removal [7-10], face restoration [11], film and television special effects production, image editing, etc. Therefore, image inpainting technology has always been one of the key issues explored by researchers in the field of computer vision [12-14].

Since the concept of image inpainting was put forward, it has aroused the interest of a large number of researchers. Using computer technology and mathematical theory knowledge, they provided solutions to various image inpainting problems and made image inpainting technology develop rapidly. There are many methods of image inpainting, and some typical image restoration methods are briefly introduced below.

*Corresponding author. Email: guohr@hpu.edu.cn

Partial differential equation theory is widely used in mathematics and is also an important tool in the field of image inpainting [15]. After applying the theory of partial differential equation to image inpainting technology, the problem to be solved is not only to complete the flat part of the image but also to inpainting the texture and main corners of the image. Bertalmio, SaPiro, Caselles, and Bellester [16] for the first time in 2000 set up four people named after the initials BSCB image inpainting model. Chan et al. [17] combined the BSCB model and the TV model and applied them to image inpainting, achieved good results. Although the method based on the partial differential equation has a certain effect, its applicability is limited. It is more suitable for inpainting images with small damaged areas and less structural loss, but poor performance on images with large damaged areas and more structural loss [18].

The image inpainting method based on exemplar patches has been widely applied in recent years. Drori et al. [19] proposed a fragment-based image inpainting method, which takes the undamaged area of the image as the training set to predict the damaged area of the image, and used the iterative method to advance step by step to the undamaged area and synthesize adaptive image fragments, achieving good results. However, this method adopts the global search method when searching the fragments with similar texture information, and the inpainting speed is relatively slow. Borikar et al. [20] proposed a faster algorithm to make the inpainting faster, but the inpainting effect is relatively poor. Based on the Efros

texture synthesis algorithm [21], Criminisi et al. [6] first proposed the priority order exemplar patches inpainting algorithm in 2004, which is the most classic texture synthesis image inpainting method. The main idea of this method is to use a priority formula composed of the item of confidence items and data items to determine the priority patches to be repaired in the region to be repaired, and then search the matching patches through the global search, determine the matching patches with the highest priority, and finally fill the matching patches with the highest priority into the area to be repaired. Based on the Criminisi method, many improvements have been made [22-26], and the improvement in the inpainting effect and time is obvious. Li et al. [27] improved the priority formula so that it would not lose its function from beginning to end. Alexander et al. [28] introduced a loss function to calculate the weight of each exemplar patch when searching for the best-matched patch, and synthesized the optimally matched patch with the global information of multiple sample patches to inpainting the damaged area [29]. Tang et al. [30] proposed an image coding framework based on texture synthesis and adopted a constrained exemplar patch search method to narrow the search scope to areas with similar textures, which has a good inpainting effect. Gong et al. [31] combined the fast marching method [32] with the texture information of images to improve the inpainting sequence of exemplar patches. Zhang et al. [33] used an improved exemplar-based inpainting method to inpainting images, introduced the level set distance
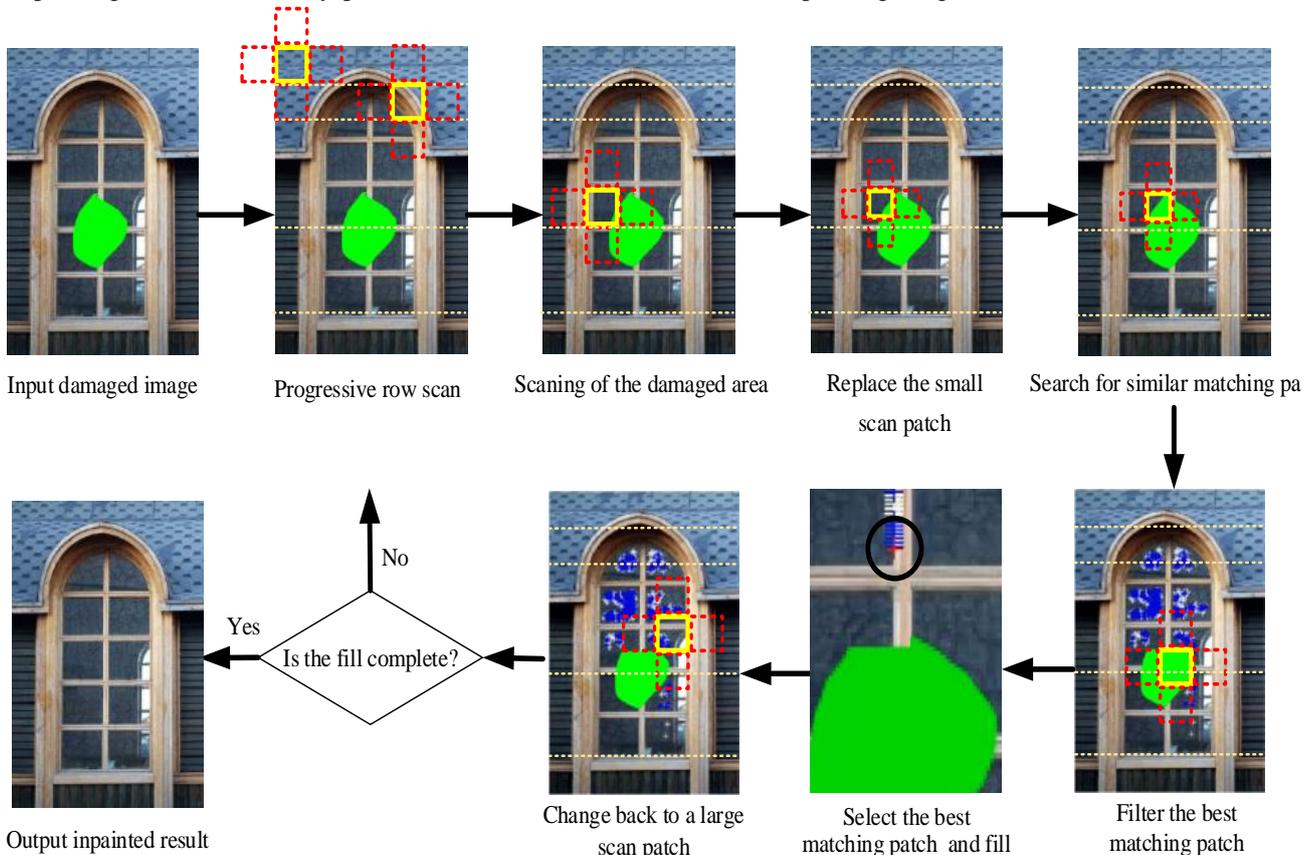

Figure 1. Algorithm flow chart

Component [34] into the priority function, and then used filters to smooth and de-noise the images, achieving better inpainting effects. Wang et al. [35] introduced regularization items to avoid the loss of priority item, and adjusted the search strategy of matching patches, thus improving the robustness of the method. Wang et al. [36] also proposed an image inpainting method based on structure tensor and adopted an improved matching patch similarity calculation method to improve the effectiveness of the algorithm.

In the inpainting method based on exemplar patches, the size of exemplar patches is fixed, which cannot adapt to the exemplar patches requirements of different images. Moreover, it adopts the global search method to search for similar matching patches, which wastes computing resources and increases the time required for inpainting. In addition, it neglects the importance of the colour information and pays too much attention to the structure information, resulting in the possibility of not matching similar patches. To solve these problems, a fast image inpainting method based on an adaptive scanning strategy is proposed in this paper. First, the image to be repaired is scanned line by line with a variable scale cross-scanning patch. A large-scale scanning patch is used for scanning for the first time. When the area to be repaired is reached, a small-scale scanning patch is adaptively changed to a big-scale scanning patch. Then the image is searched in a small area of the patch to be repaired, and the search scope is enlarged from small to large. Then the weight similarity calculation formula is used to search for similar matching blocks. Finally, in order to further improve the correctness of the selection of matching patches, multilevel matching criteria and screening strategies are used to screen and then fill. The algorithm flow is shown in Figure 1.

2. Related work

2.1. Symbolic representation

As shown in Figure 2, Ω is the known region in the damaged image entered, Φ is the damaged area, and Ψ_x is a square scanning patch with x as the central and 9×9 in size, which
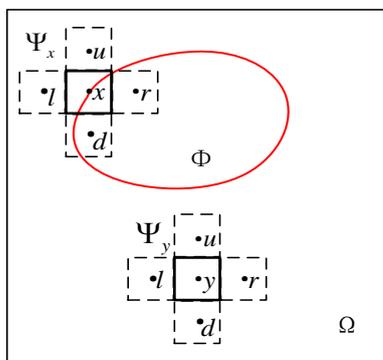

Figure 2. Symbolic representation

is called the central scanning patch. Ψ_u, Ψ_d, Ψ_l, and Ψ_r are square patches of the same size and adjacent to Ψ_x located around Ψ_x. Ψ_x, Ψ_u, Ψ_d, Ψ_l, and Ψ_r together form a cross-scanning patch, and Ψ_y is a matching patch similar to Ψ_x.

2.2. Introduce the structure tensor

Let the structure tensor of any point q of the image be $P(q)$, $P(q)$ is a symmetric semi-positive definite two-dimensional matrix. Its two eigenvalues t_1 and t_2 are used to define the correlation of image data. The edge intensity E at pixel $q(i, j)$ can be defined as

$$E(i, j) = (t_1 - t_2)^2 \qquad (1)$$

The angle intensity M at pixel point $q(i, j)$ is:

$$M(i, j) = (t_1 \times t_2) / (t_1 + t_2) \qquad (2)$$

The edge intensity E and angle intensity M can measure the anisotropy degree of the edge structure features in the region around the pixel, and the eigenvalues t_1 and t_2 can be used to analyse the local structure of the image.

The local structural features of the image are discussed by using eigenvalues as follows:

(1) When $t_1 \approx t_2 \approx 0$, the edge intensity $E(i, j) \approx 0$ and the angle intensity $M(i, j) \approx 0$, indicating that the grey value of the image in all directions near this pixel point has little change, namely, it is a flat region;

(2) When $t_1 \gg t_2 \approx 0$, the edge intensity $E(i, j) \gg 0$ and the angle intensity $M(i, j) \approx 0$, it shows that the grey value of the image has a large change rate along a certain direction, that is, there is an edge structure;

(3) When $t_1 \approx t_2 \gg 0$, the edge intensity $E(i, j) \approx 0$ and the angle intensity $M(i, j) \gg 0$, this indicates that the grey value of the image changes rapidly in two mutually perpendicular directions, that is, there is an angle structure;

The above discussion of eigenvalues can be used to correctly identify the situation near a certain pixel point, to match the correct pixel patch.

2.3. Edge and angle intensity model

Here, edge intensity and corner intensity are used to distinguish the features of different structures in the image. By calculating the edge and angle intensity of blocks Ψ_x and Ψ_y to be matched, the accuracy of structural information in the target image can be improved.

To improve the quality of image inpainting, this paper uses the edge and angle intensity model to identify the consistency

of patches Ψ_x and Ψ_y to be matched. Formula (3) is the constructed edge and angle intensity model:

$$\bar{Z}(x,y) = \frac{(E(i_x,j_x) + E(i_y,j_y))^2 + (M(i_x,j_x) + M(i_y,j_y))^2}{\varepsilon} \quad (3)$$

Here $\varepsilon = 0.001$ (experience points).

According to the characteristics of t_1 and t_2 of the two eigenvalues of the structure tensor and formula (3), if $\bar{Z}(x,y)$ is equal to or approximately 0, it indicates that the correlation between patch Ψ_x and patch Ψ_y to be matched is relatively low, and this pixel is located in a flat region. If $\bar{Z}(x,y)$ is

much larger than 0, it stands that the correlation between patch Ψ_x and patch Ψ_y to be matched is relatively high, and there is more structural information around the pixel point, such as edge structure or angle structure.

The edge and angle intensity model is crucial for inpainting the edge and angle structure in images. In Figure 3, we can observe the difference in image inpainting results when the proposed algorithm is used with and without the edge and angle intensity model. Without this model, the structural information of the image cannot be effectively repaired, leading to incorrect pixel patches. On the other hand, the edge and angle intensity model improves the accuracy of inpainting by inpainting the structure and texture information of the image without any errors in pixel patches.
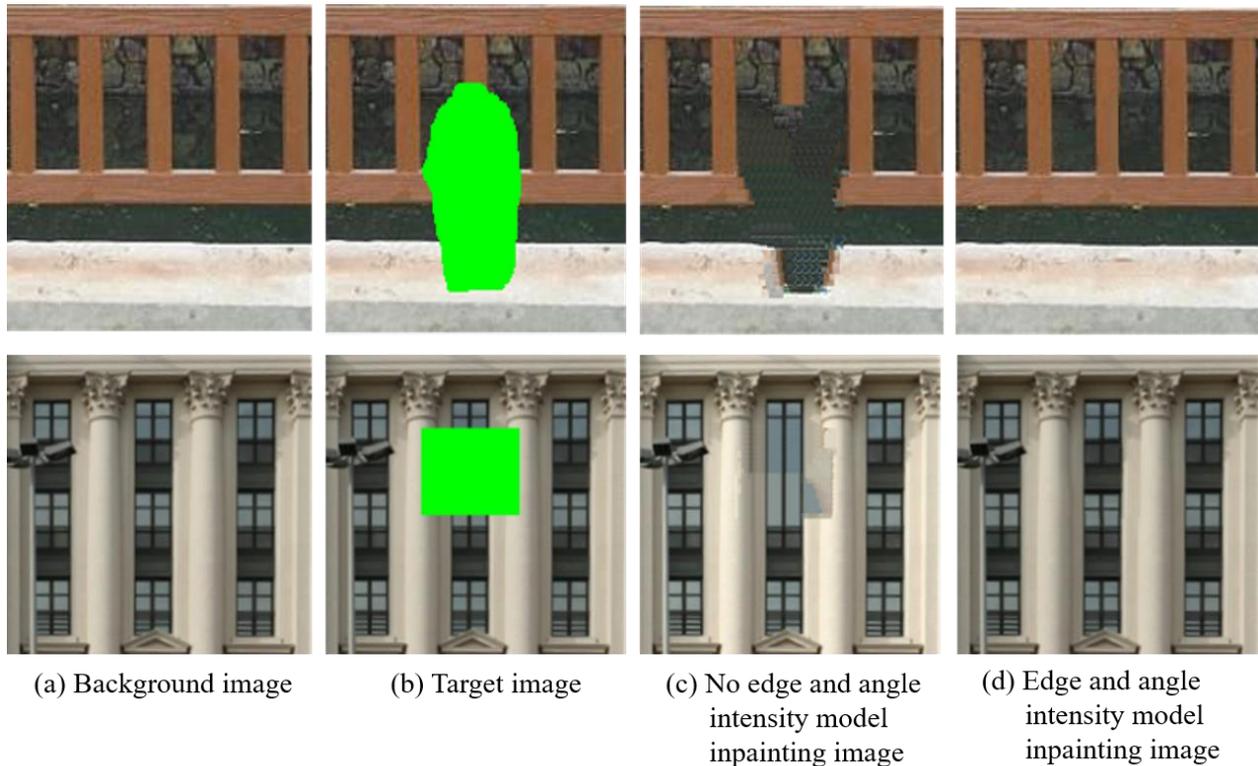

(a) Background image   (b) Target image   (c) No edge and angle intensity model inpainting image   (d) Edge and angle intensity model inpainting image

Figure 3. Edge and angle intensity model inpainting experiment

3. Arithmetic Design

3.1. Progressive row scan

To avoid the problem of wrong inpainting order and low inpainting efficiency caused by the priority function in the exemplar patch-based inpainting algorithm, this paper adopts the progressive line scanning method with an adaptive scanning patches scale. In this method, a cross-scanning patch of 15×15 pixel size is used to scan from the

upper left corner of the input damaged image in the order of left to right and top to bottom. The cross-scan patch moves 15 pixels at a time, scanning the first line and then the following line. With the progress of scanning, when the cross-scanning patch scans the green area to be repaired, the small-scale scanning patch with the size of 9×9 is replaced and continues to scan, moving 1 pixel each time. When the centre scan patch Ψ_x and the region Φ to be repaired have the maximum overlapping area and satisfy formula (4), that is, there is no overlapping area between the region Φ to be repaired and the upper patch Ψ_u and the right patch Ψ_l of the centre scan patch, the search for

similar matching patches begins. When the area to be repaired cannot be scanned, large-scale scanning patches are replaced. For details, please refer to the algorithm flow chart in figure 1.

$$\Phi \cap (\Psi_u \cup \Psi_l) = \varnothing \qquad (4)$$

3.2. Search range

Since the image inpainting algorithm based on exemplar patches adopts the global search for similar matching patches, the complexity of the algorithm is very high. Each search requires the similarity function to calculate the similarity and carry out screening. As a result, the whole algorithm takes a long time. However, an image often has local self-similarity. If a global search is used to search for similar patches, computing resources will be wasted, and the best matching patches are still found in the neighbourhood of matching patches. Therefore, this paper proposes a new local search strategy to improve inpainting efficiency. The local search strategy is as follows:

When the scanning patch scans the damaged area, it takes the damaged point pixel as the centre and diffuses around to determine the local search area P, as shown in Figure 4:
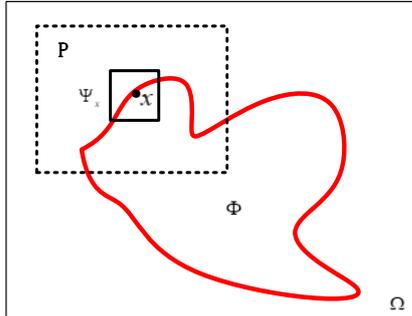

Figure 4. Local search scope diagram

Where Ω is the image to be repaired with the size $m \times n$, Φ is the area to be repaired, Ψ_x is the block to be matched with x as the centre. P is a rectangle with x as the centre and the size of $(m/k, n/k)$. k is variable and is taken as 3 for the first time. When the best matching block is found, the next step will be continued. If the best matching patch is not found, k will be taken as 2, and finally 1, that is, the search will be performed globally. Using the local similarity of image patches and changing the search range can effectively improve the inpainting speed of the algorithm. After searching with this method, the matching patches' similarity needs to be calculated. The following describes the weight similarity calculation formula.

3.3. Similarity calculation

In traditional matching patch search strategies, few factors are considered, such as only colour features or single structure features, here, the edge and angle intensity model and SSD are combined to identify the colour and structural features of the image. The weight similarity calculation formula is required to calculate the similarity between the matching blocks and the patches to be matched. The weight similarity calculation formula is defined as follows:

$$\bar{S}(\Psi_x, \Psi_y) = \mu D(\Psi_x, \Psi_y) + \eta \times \bar{Z}(x, y) \qquad (5)$$

Where μ and η are constants. Through a large number of experiments, $\mu = 3$, $\eta = 9$, $\bar{Z}(x, y)$ is the edge and angle intensity model, and $D(\Psi_x, \Psi_y)$ is the SSD distance formula, expressed as:

$$D(\Psi_x, \Psi_y) = \Sigma \sqrt{(R_{\Psi_x} - R_{\Psi_y})^2 + (G_{\Psi_x} - G_{\Psi_y})^2 + (B_{\Psi_x} - B_{\Psi_y})^2} \qquad (6)$$

Here R, G, and B mean the pixel corresponding to the red, green, and blue channels.

In this paper, the known information in the cross-scan patch can be used to calculate the edge and angle intensity, accurately identify the texture structure information around the block Ψ_x to be filled, and then find similar matching patches in the image according to formula (5). To improve the accuracy of matching, priority matching criteria need to be used for matching.

3.4. Six-level matching criteria

The change in image gradient can reflect the change of local structure information of the image, but the gradient of a single pixel cannot be reflected. Therefore, the pixel adjacent to the centre pixel can be used to identify the local texture structure more accurately. To correctly reconstruct the local structure information near the area to be repaired and improve the robustness of the method, an inpainting factor model is proposed here. The inpainting factor model is defined as:

$$\bar{G} = \frac{(G_1 + G_2 + \cdots + G_x + \cdots + G_{n-2} + G_{n-1})}{n} \qquad (7)$$

Where, G_x is the gradient value at the centre pixel point x, and G_1, G_2, G_{n-1} and G_{n-2} are the gradient values of the pixel points in the four directions of the top, bottom, left, and right adjacent to the centre pixel point x, where n is taken as 5.

The local texture structure can be more accurately identified by using the inpainting factor, because there may be more than one similar matching patch searched by formula (5) in the global, so it is necessary to further filter the matching patches. The following defines the filtering strategy for filtering the matching patches:

$$\Psi_{\bar{y}} < \arg\min_{\Psi_y \in \Omega} S(\Psi_x, \Psi_y) \times f \qquad (8)$$

Where $S(\Psi_x, \Psi_y)$ is the weight similarity function of formula (5), f is the error correction factor, and the error correction factor f is defined as follows:

$$f = \gamma + \bar{G} \times (E(i,j) + M(i,j)) \qquad (9)$$

Where $\gamma = 1.3$ (empirical value), $\bar{G}$ is the inpainting factor model, $E(i,j)$ is the edge intensity of the block to be matched Ψ_x with the central pixel point of $x(i,j)$, and $E(i,j)$ is the angle intensity of the patch to be matched with the central pixel point of $x(i,j)$.

After filtering, the matching patch Ψ_y may have the following six situations:

(i) If the matching patch Ψ_y and its surrounding patches Ψ_u, Ψ_d, Ψ_l, and Ψ_r match the image information of the corresponding patches of the patch Ψ_x to be matched, it is the best matching patch, and the matching priority is level Ⅰ;

(ii) If four patches match the image information of matching patch Ψ_y and its patches Ψ_u, Ψ_d, Ψ_l, and Ψ_r and corresponding patch Ψ_x, so the matching priority of matching patch Ψ_y is level Ⅱ.

(iii) If three patches match the image information of matching patch Ψ_y and its patches Ψ_u, Ψ_d, Ψ_l, and Ψ_r and corresponding patch Ψ_x, so the matching priority of matching patch Ψ_y is level Ⅲ.

(iv) If two patches match the image information of matching patch Ψ_y and its patches Ψ_u, Ψ_d, Ψ_l, and Ψ_r and corresponding patch Ψ_x, so the matching priority of matching patch Ψ_y is level Ⅳ.

(iv) If only one patch match the image information of the patch Ψ_y and its surrounding patches Ψ_u, Ψ_d, Ψ_l, and Ψ_r with that of the patch Ψ_x to be matched, the matching priority is level Ⅴ;

(vi) If the image information of the matching patch Ψ_y and its surrounding patches Ψ_u, Ψ_d, Ψ_l, and Ψ_r do not match that of the corresponding patch Ψ_x, the matching priority is level Ⅵ;

The selected matching patches have six priorities, with level Ⅰ being the highest and level Ⅵ the lowest. The matching patches are matched according to their priorities. The patch with the highest priority is the best matching patch, and the pixels in the best matching patch are copied and filled into the patch to be matched.

3.5. Algorithmic pseudocode

The pseudo-code of this algorithm is shown as follows:
　Input: Marked colour of the image to be repaired;
　Output: Repaired image;

Step 1: The central scanning patch Ψ_x of size 15×15 is used as the basic scanning patch, and the repaired image is progressively scanned from left to right and from top to bottom to find the area Φ to be repaired. When the cross-scanning patch scans the green area to be repaired, the small-scale scanning patch with the size of 9×9 will be replaced and the scanning will continue until the central scanning patch Ψ_x and the area to be repaired have the maximum overlap, and the similar matching patches will be searched.

Step 2: While $\Phi \cap (\Psi_u \cup \Psi_l) = \varnothing$;

Do

(1) The edge and angle intensity model is constructed, and the weight similarity formula (5) is used to search the image. First, a search is conducted within the range of $(m/3, n/3)$. If no similar matching patches can be found, the search scope is expanded to $(m/2, n/2)$ until candidate matching patches for the patch Ψ_x are found throughout the entire image.

(2) Formula (8) is used to preliminarily screen the candidate patches in step (1), and then the six-level priority matching criterion is used to further screen the candidate patches;

(3) When the area to be repaired cannot be scanned, the large-scale scanning patch of 15×15 is replaced to continue scanning;

End;

Sept 3: Repeat these steps until the image inpainting is complete.

4. Experimental results

Image inpainting is an estimation technique for damaged information. The visual subjective evaluation of image inpainting results is susceptible to the objective influence of individuals. If some fine texture or structure information is judged by subjective vision, the results will vary from person to person. Peak signal-to-noise ratio (PSNR), structural similarity (SSIM), and inpainting time can reflect the difference in the inpainting effect of different algorithms. Therefore, the three evaluation indexes were selected in this paper to compare the inpainting results of methods [6], [27], and [35]. To verify the effectiveness of the method presented in this paper, three groups of experiments, namely damaged image inpainting, texture inpainting, and target removal, are carried out in this section. The following three groups of experiments are described respectively.
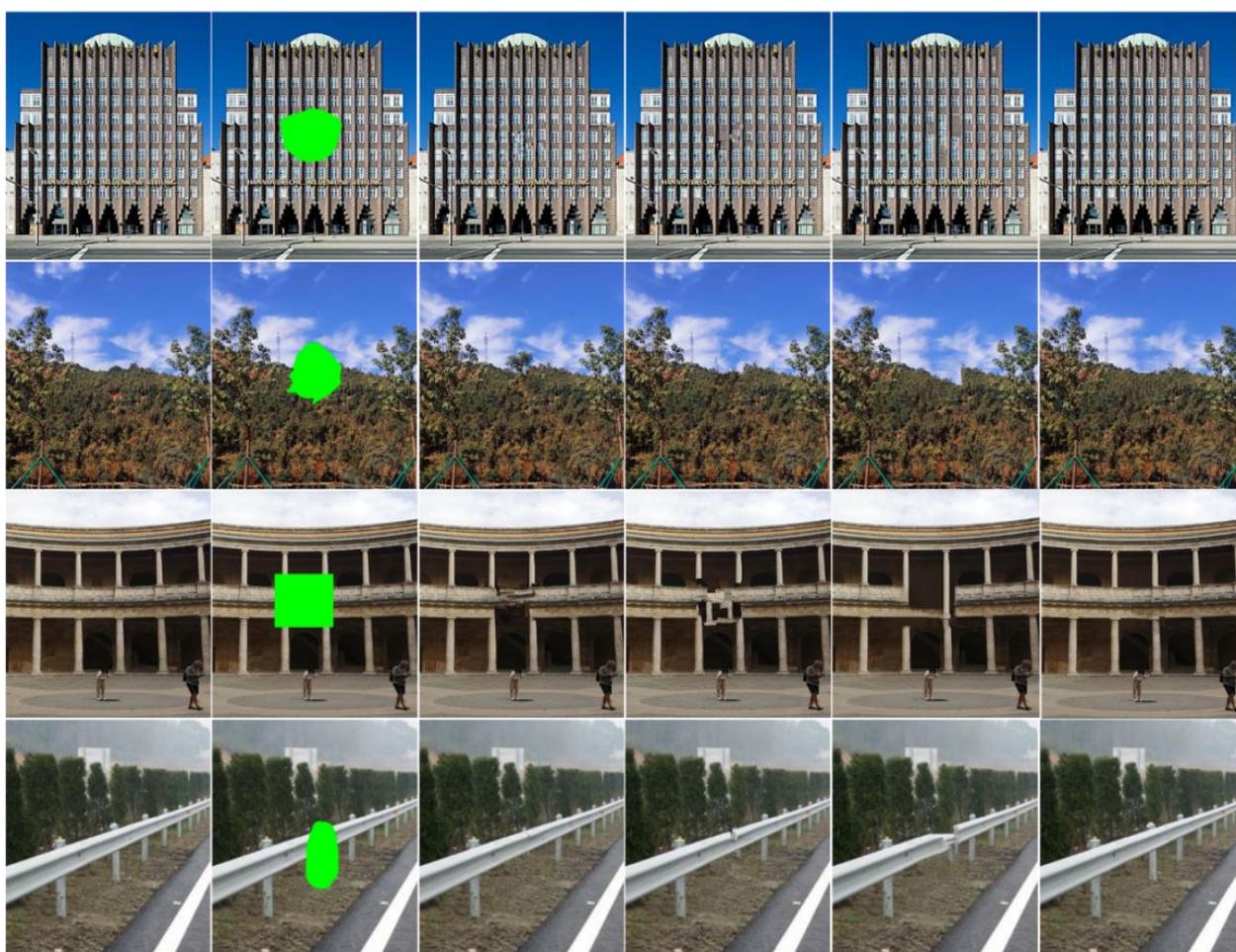
4.1. Damaged image inpainting experiment

This section uses images from four different environments to test the effectiveness of the algorithm proposed in this paper. As shown in Figure 5, column (a) shows the original image, column (b) shows the original image with randomly

added masks of different shapes, column (c) shows the inpainting results of the method used in reference [6], column (d) shows the inpainting results of the method used in reference [27], column (e) shows the inpainting results of the method used in reference [35], and column (f) shows the inpainting results of the method proposed in this paper.

As can be seen from the first line of the image in Figure 5, there are some wrong pixels in the inpainting results of the method of literature [6], literature [27], and literature [35], for example, the pixel of the window is mistakenly copied to the wall or the pixel of the wall is mistakenly copied to the window. As can be seen from the images in the second line, the inpainting results of literature [6], literature [27], and literature [35] all show the unnatural connection of pixel patches, while the algorithm in this

paper has a better visual effect. In the third row image, the inpainting results of the literature [6] method, literature [27] method, and literature [35] method also showed the breakage of pixel patches, and a large number of pixel patches were wrongly filled in the area to be repaired, resulting in poor inpainting results. The fourth-row image is the inpainting of the fence. The method of literature [6], literature [27], and literature [35] failed to repair the fence posts, and the situation of pixel patch fracture also appeared. The reason for this situation in the methods of literature [6], literature [27] and literature [35] is that the colour and structural characteristics of the area to be repaired are not fully considered, or the strategy of screening matching patches is not good.

(a) Original image   (b)Mask image   (c)Method in [6]   (d) Method in [27]   (e) Method in [35]   (f)Our method

Figure 5. Damaged image inpainting experiment
Table 1. PSNR of damaged image inpainting by different algorithms

Image	Method in [6]	Method in [27]	Method in [35]	**Our method**
Figure 5 first line	32.9639	33.0200	32.9032	**33.3309**

Figure 5 second line	30.9837	33.8667	31.8065	**34.6214**
Figure 5 third line	30.4969	29.5343	30.5333	**35.6836**
Figure 5 fourth line	37.0907	36.6428	32.4437	**38.9610**

Table 2. SSIM of damaged image inpainting by different algorithms

Image	Method in [6]	Method in [27]	Method in [35]	**Our method**
Figure 5 first line	0.9841	0.9830	0.9771	**0.9892**
Figure 5 second line	0.9687	0.9745	0.9699	**0.9751**
Figure 5 third line	0.9681	0.9618	0.9728	**0.9772**
Figure 5 fourth line	0.9833	0.9825	0.9322	**0.9834**

Table 3. The running time of different algorithms for inpainting damaged images

Image	Method in [6]	Method in [27]	Method in [35]	**Our method**
Figure 5 first line	22.169	21.376	23.546	**14.651**
Figure 5 second line	41.128	46.100	22.877	**21.364**
Figure 5 third line	45.765	49.815	31.907	**26.956**
Figure 5 fourth line	17.619	18.770	13.246	**12.428**

Table 1, Table 2 and Table 3 respectively show the PSNR value, SSIM value, and inpainting time of the inpainting results of various methods in the damaged image inpainting experiment in Figure 5. It can be seen from Figure 5 and these three tables that the inpainting effect and running speed of the algorithm proposed in this paper are superior to the other three algorithms.

4.2. Texture inpainting experiment

Texture image has many characteristics, such as such as repeated local pixel sequences, non-random arrangement, in a part of the scope of most of the uniform unity. Unlike from the colour feature and structure feature, the texture feature is not based on pixel point, but can reflect the global feature. When inpainting images, efficient use of texture information in images can better inpainting images.

In Figure 6, four images with different textures are used to test the ability of the proposed algorithm in inpainting texture images. As can be seen from the first line in the figure, the inpainting results of literature [6], literature [27] and, literature [35] all have many wrong pixel patches. Although some wrong pixel patches exist in the method in this paper, the overall effect is better than the other three methods. The method of literature [6] and literature [27] have a good effect on the restoration of the blue background black texture image of the second line, but there are a few wrong pixels, so it looks messy, while the method of literature [35] has a large blur. When the three

methods were used to inpainting the cracked and dry land image in the third row, the lines were disarranged, and the effect was not very ideal. When inpainting the brick wall texture, the method of literature [6], literature [27], and literature [35] all have the situation of copying pixel patches incorrectly. The method presented in this paper shows good robustness when inpainting the four texture images.

Table 4, Table 5 and Table 6 respectively show the PSNR value, SSIM value, and inpainting time of the four methods' inpainting results in the texture inpainting experiment in Figure 6. It can be seen from Figure 6 and these three tables that the inpainting effect and speed of the method proposed in this paper are superior to the other three methods.

4.3. Target removal experiment

Target removal refers to removing the selected area in the image and reconstruct the area according to the existing information in the image. Unlike the previous two groups of experiments, there is no real background map that can be quantitatively analysed. Therefore, as there is no established standard for target removal experiments, only subjective visual analysis can be used to evaluate the quality of inpainting. Figure 7 shows the renderings of object removal in different scene images by the method of literature [6], literature [27], literature [35], and this paper. The stone pot is removed from the figure in the first row.

None of the four methods can completely reconstruct reasonable pixels, but the algorithm in this paper has a better inpainting effect. In the second row, green plants on the island were removed. Both the method in literature [6] and the method in literature [27] incorrectly reconstructed the water area in the sky, while the method in literature [35] missed some pixels of the beach. The figure in the third row was removed. The method of literature [6] and literature [27] incorrectly reconstructed the pixels of the upper tree in the rapeseed field, while the method of literature [35] missed some pixels and the connection was not natural. In the fourth row, people on the path were removed, the method of literature [6] had redundant grass, the method of literature [27] had many wrong pixel patches, and the

method of literature [35] widened the normal road. The experimental effect of object removal on the four images presented in this paper is better and more in line with visual perception.

To verify the effectiveness of the method presented in this paper, experiments of damaged image inpainting, texture inpainting, and target removal are carried out in this section, and the results of image inpainting are verified from the perspectives of qualitative and quantitative analysis. Through the simulation experiments on several groups of images with different structure types, the results show that the proposed method is superior to the algorithms in literature [6], [27] and, [35], and has higher inpainting speed and stronger robustness.
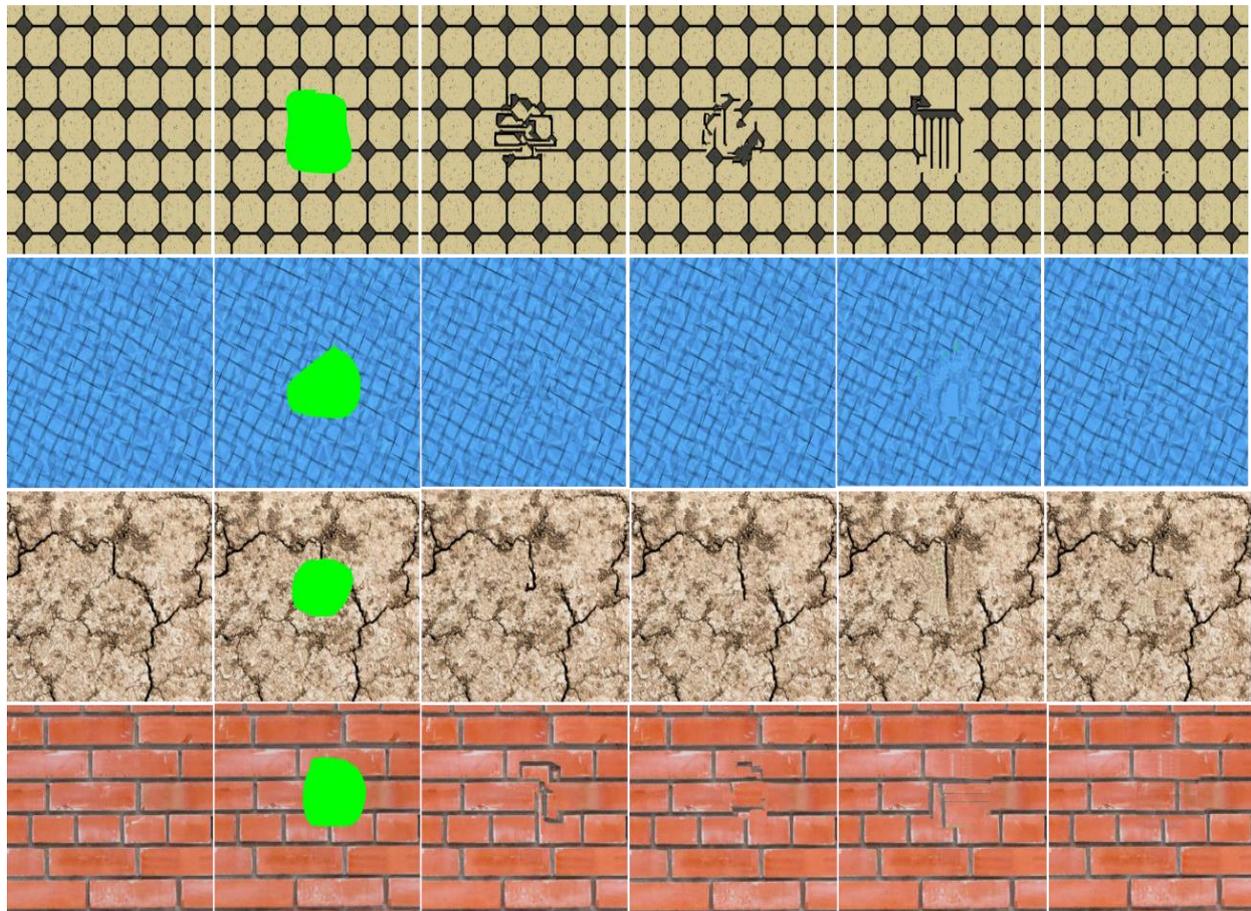
(a) Original image   (b)Mask image   (c)Method in [6]   (d) Method in [27] (e) Method in [35]   (f)Our method

Figure 6. Texture image inpainting experiment
Table 4. PSNR of texture image inpainting by different algorithms

Image	Method in [6]	Method in [27]	Method in [35]	**Our method**
Figure 6 first line	23.7707	25.0439	23.8356	**31.3771**
Figure 6 second line	35.8307	36.2405	34.7355	**36.5459**
Figure 6 third line	29.7008	29.7974	29.9351	**30.8044**
Figure 6 fourth line	31.6733	33.1288	31.9683	**34.7478**

Table 5. SSIM of texture image inpainting by different algorithms

Image	Method in [6]	Method in [27]	Method in [35]	Our method
Figure 6 first line	0.9382	0.9469	0.8852	**0.9652**
Figure 6 second line	0.9947	0.9950	0.9939	**0.9954**
Figure 6 third line	0.9607	0.9605	0.9592	**0.9682**
Figure 6 fourth line	0.9756	0.9817	0.9846	**0.9856**

Table 6. The running time of different algorithms for inpainting texture images

Image	Method in [6]	Method in [27]	Method in [35]	Our method
Figure 6 first line	133.10	142.20	32.191	**30.235**
Figure 6 second line	37.726	42.303	44.251	**36.845**
Figure 6 third line	25.550	29.798	21.832	**18.260**
Figure 6 fourth line	50.393	43.300	59.280	**38.971**
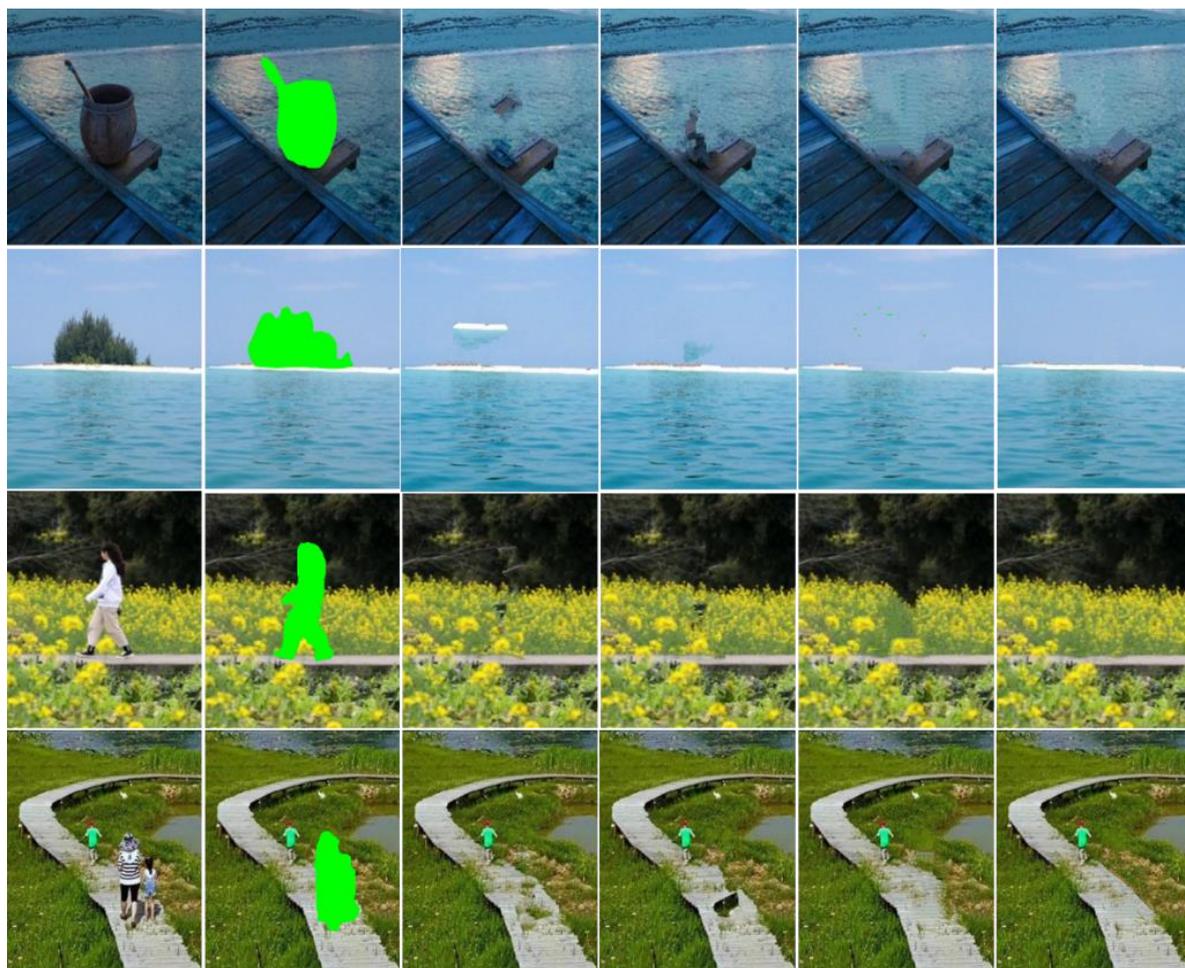
(a) Original image   (b)Mask image   (c)Method in [6] (d) Method in [27] (e) Method in [35]   (f)Our method

Figure 7. Target removal experiment

5. Conclusions

In this paper, a fast image inpainting method based on an adaptive scanning strategy is proposed. We use scanning patches with variable scales for line progressive scanning. First, large-scale scanning patches are used for scanning, and when the area to be repaired is scanned, the adaptive change is made to a small-scale scanning patch. An edge and angle intensity model has been constructed, and a weight similarity function was used to search for similar matching patches to identify the local structure of the image, which solved the problem that the inpainting failed due to ignoring the importance of angle structure and colour information. In addition, according to the local self-similarity of the image, the method of variable search scope is used to improve the inpainting speed. Firstly, the search scope of similar matching patches in a small range is changed from small to large, which saves computing resources. The correctness of matching patches is ensured by using six-level priority matching criteria and screening criteria. Finally, three sets of experiments were conducted to verify that PSNR, SSIM, and inpainting time were superior to the other three methods, indicating that the proposed algorithm has higher robustness and inpainting speed.

References

[1] Buyssens P, Daisy M, Tschumperle D, et al. Exemplar-based Inpainting: Technical Review and new Heuristics for better Geometric Reconstructions [J]. IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society, 2015, 24(6): 1809-24.

[2] Yudong Zhang, Pattern Recognition via PCNN and Tsallis Entropy, Sensors, 2008, 8(11):7518-7529.

[3] T. K. Shih, Rong-Chi Chang, Liang-Chen Lu and Huan-Chi Huang, "Multi-layer inpainting on Chinese artwork [restoration applications]," 2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No.04TH8763), Taipei, Taiwan, 2004, pp. 21-24.

[4] T. K. Shih and Rong-Chi Chang, "Digital inpainting - survey and multilayer image inpainting algorithms," Third International Conference on Information Technology and Applications (ICITA'05), Sydney, NSW, Australia, 2005, pp. 15-24 vol.1, doi: 10.1109/ICITA.2005.169.

[5] A. Criminisi, P. Perez and K. Toyama, "Object removal by exemplar-based inpainting," 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings., Madison, WI, USA, 2003, pp. II-II, doi: 10.1109/CVPR.2003.1211538.

[6] Criminisi A, Perez P, Toyama K. Region Filling and Object Removal by Exemplar-Based Image Inpainting [J]. IEEE Transactions on Image Processing, 2004, 13(9): 1200-12.

[7] Shui-Hua Wang, Unilateral sensorineural hearing loss identification based on double-density dual-tree complex wavelet transform and multinomial logistic regression, Integrated Computer-Aided Engineering, 2019, 26: 411-426.

[8] S. Kumar, M. Biswas, S. J. Belongie and T. Q. Nguyen, "Spatio-temporal texture synthesis and image inpainting for video applications," IEEE International Conference on Image Processing 2005, Genova, Italy, 2005, pp. II-85, doi: 10.1109/ICIP.2005.1529997.

[9] Li-Qin J I， Wang J J . Automatic Text Detection and Removal in Video Images[J]. Journal of Image and Graphics, 2008, 13(3):461-466.

[10] M. Khodadadi and A. Behrad, "Text localization, extraction and inpainting in colour images," 20th Iranian Conference on Electrical Engineering (ICEE2012), Tehran, Iran, 2012, pp. 1035-1040, doi: 10.1109/IranianCEE.2012.6292505.

[11] Elharrouss O, Almaadeed N, Al-Maadeed S, et al. Image inpainting: A review[J]. Neural Processing Letters, 2020, 51: 2007-2028.

[12] Yudong Zhang, A seven-layer convolutional neural network for chest CT-based COVID-19 diagnosis using stochastic pooling, IEEE Sensors Journal, 2022, 22(18): 17573 – 17582.

[13] Wang M. Movie2Comics: Towards a Lively Video Content Presentation [J]. IEEE Transactions on Multimedia, 2012, 14(3): 858-70.

[14] Huang J B, Kang S B, Ahuja N, et al. Temporally Coherent Completion of Dynamic Video [J]. ACM Transactions on Graphics, 2016, 35(6cd): 196.

[15] Boscain U V, Chertovskih R, Gauthier J-P, et al. Highly corrupted image inpainting through hypoelliptic diffusion [J]. Journal of Mathematical Imaging and Vision, 2018, 60: 1231-45.

[16] Boscain U V, Chertovskih R, Gauthier J P, et al. Highly corrupted image inpainting through hypoelliptic diffusion[J]. Journal of Mathematical Imaging and Vision, 2018, 60: 1231-1245.

[17] Chan T, Shen J. Local inpainting models and TV in-painting [J]. 2001.

[18] Yu-Dong Zhang, MIDCAN: A multiple input deep convolutional attention network for Covid-19 diagnosis based on chest CT and chest X-ray, Pattern Recognition Letters, 2021, 150: 8-16.

[19] Drori I, Cohen-Or D, Yeshurun H. Fragment-based image completion [J]. Acm Transactions on Graphics, 2003, 22(3): 303-12.

[20] Billett S. Fast Algorithm for Completion of Images with Natural Scenes [J]. Mind Culture & Activity, 2004.

[21] Efros A A, Freeman W T. Image quilting for texture synthesis and transfer[C]//Proceedings of the 28th annual conference on Computer graphics and interactive techniques. 2001: 341-346.

[22] Barnes C, Shechtman E, Finkelstein A, et al. PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing [J]. ACM Trans Graph, 2009, 28(3, article 24).

[23] Shui-Hua Wang, Diagnosis of COVID-19 by wavelet Renyi entropy and three-segment biogeography-based optimization, International Journal of Computational Intelligence Systems, 2020: 13(1): 1332-1344.

[24] Zhang C F, Shao L P, Shi J. Improved Criminisi Algorithm Based on Adaptive Gradient Classification and Matching [J]. Journal of Chinese Computer Systems, 2018.

[25] Tao Z, Zhang J, Wang L, et al. Image inpainting algorithm based on edge feature and pixel structure similarity [J]. Journal of Computer-Aided Design & Computer Graphics, 2019, 31(10): 1768-76.

[26] Qiang Z, He L, Chen X, et al. Image Inpainting Using Image Structural Component and Patch Matching [J]. Jisuanji Fuzhu Sheji Yu Tuxingxue Xuebao/Journal of Computer-Aided Design and Computer Graphics, 2019, 31(5): 821.

[27] Li X F, Wang J, Liu H M, et al. Image inpainting using feature precedence and patch matching[J]. Journal of Computer-Aided Design, 2016: 1131-1137.

[28] A. Wong and J. Orchard, "A nonlocal-means approach to exemplar-based inpainting," 2008 15th IEEE International Conference on Image Processing, San Diego, CA, USA, 2008, pp. 2600-2603, doi: 10.1109/ICIP.2008.4712326.

[29] Shuihua Wang, Advances in data preprocessing for biomed ical data fusion: an overview of the methods, challenges, a nd prospects, Information Fusion, 2021, 76: 376-421.

[30] Tang F, Ying Y, Wang J, et al. A novel texture synthesis based algorithm for object removal in photographs; proceedings of the Advances in Computer Science-ASIAN 2004 Higher-Level Decision Making: 9th Asian Computing Science Conference Dedicated to Jean-Louis Lassez on the Occasion of His 5th Birthday Chiang Mai, Thailand, December 8-10, 2004 Proceedings 9, F, 2005 [C]. Springer.

[31] Gong X, Liu J, Zhou W, et al. Guided depth enhancement via a fast marching method [J]. Image & Vision Computing, 2013, 31(10): 695-703.

[32] Telea A. An Image Inpainting Technique Based on the Fast Marching Method [J]. Journal of Graphics Tools, 2004, 9(1): 23-34.

[33] L. Zhang, P. Shen, S. Zhang, J. Song and G. Zhu, "Depth enhancement with improved exemplar-based inpainting and joint trilateral guided filtering," 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 2016, pp. 4102-4106, doi: 10.1109/ICIP.2016.7533131.

[34] Jing W, Ke L, Pan D, et al. Robust object removal with an exemplar-based image inpainting approach [J]. Neurocomputing, 2014, 123(jan.10): 150-5.

[35] Wang J, Zhou Y H, Sima H F, et al. Image Inpainting Based on Structural Tensor Edge Intensity Model[J]. International Journal of Automation and Computing, 2021, 18(2): 256-265.

[36] He Y, Tang X, Zhang Y, et al. Improved Criminisi algorithm based on structure tensor [J]. Journal of Image and Graphics, 2018.