IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received April 26, 2020, accepted May 15, 2020, date of publication May 19, 2020, date of current version June 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2995700

Image Inpainting for Object Removal Based on Adaptive Two-Round Search Strategy

LEI ZHANG AND MINHUI CHANG
School of Mathematics and Information Technology, Yuncheng University, Yuncheng 044000, China

Corresponding author: Lei Zhang (zhanglei@ycu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61703363, in part by the Scientific and Technological Innovation Programs of Higher Education Institutions in Shanxi under Grant 2019L0855, in part by the Industrial Science and Technology Research Project of Henan Province under Grant 202102210387, and in part by Scientific Research Project of Yuncheng University under Grant YQ-2017027, Grant XK-2018034, and Grant CY-2019025.

ABSTRACT In traditional inpainting method for object removal, the SSD (Sum of Squared Differences) is always used to measure the degree of similarity between exemplar patch and target patch. Although the matching rule is simple, there is a risk that the target patch is replaced by an unsuitable exemplar patch, which leads to the mismatch error. Even worse, the error may be constantly accumulated along with the process progresses, finally some unexpected objects will be introduced into target region, and the restored image cannot meet the requirements of human vision. In view of these problems, we propose an inpainting method based on adaptive two-round search strategy. Firstly, we define the DBP (Differences Between Patches) between target patch and exemplar patch, and use it to measure the degree of difference between the two patches. Then, based on SSD and DBP, we adaptively judge whether there is a mismatch error. If the mismatch error occurs, the two-round search strategy is implemented. We define a new matching rule and use it to re-search the exemplar patch. Finally, we use the exemplar patch to restore the target patch. Experimental results demonstrate the effectiveness of our method. It can effectively prevent the occurrence of mismatch error and error accumulation, improve the restoration effect.

INDEX TERMS Image inpainting, object removal, two-round search strategy, mismatch error.

I. INTRODUCTION

As one of the most important branches of image processing and pattern recognition, image inpainting has attracted more and more researchers' attention recently [1], [2]. Its basic idea is to use the effective information in the undamaged regions to estimate and fill the damaged regions according to certain rules, making the restored image more natural, and making the person who is not familiar with the original image cannot notice the restoration traces [3]. At present, image inpainting technology is playing an increasingly important role in many fields [4], such as restoration of old photos and precious historical literature materials, protection of cultural relics [5], film and television special effect production, robot vision, and so on [6], [7].

Up to now, according to the basic idea, existing inpainting methods can be divided into three categories [8]: the PDE-based (Partial Differential Equation) method, the exemplar-based method, and the sparse-representation-based method.

The most fundamental method is based on the PDE [9], [10]. Its basic idea is that the effective information around the damaged region is smoothly propagated into the damaged region along the direction of the isophote, so as to restore the missing region in an unnoticeable way [11]. In 2000, Bertalmio *et al.* [12] first put forward the notion of image inpainting and proposed the BSCB model. Chan and Shen [13] proposed TV (Total Variance) model, then proposed CDD (Curvature Driven Diffusions) model [14] to solve the connectivity problem in TV model. These methods can achieve a convincing effect in restoring the small-scale damaged regions, such as removing scratches, removing text coverage, filling missing pixels, and so on. However, they may result in over-smooth phenomenon when restoring the large-scale damaged regions, such as removing objects from images.

The second category is exemplar-based method [15], [16]. Its basic idea is searching for the most similar exemplar

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Liu.

patch in source region and using it to replace the damaged patch. The most representative method is proposed by Criminisi *et al.* [17]. First of all, the priority value of each patch which is located on the boundary of damaged region is calculated, and the patch with the highest priority value is to be restored first. Then according to the matching rule, the most similar patch is found in the undamaged region. Finally, the current damaged patch is replaced by the similar patch. These steps are circulated until the whole damaged region is restored. Compared with the PDE-based method, this category of method can not only reduce the restoration time, but also prevent the over-smooth phenomenon in restoring the large-scale damaged region, and maintain the integrity and continuity of the texture. However, these methods still have some problems [18], [19], such as unreasonable filling order, mismatch error, error accumulation, greedy search, and so on.

The third category is sparse-representation-based method [20], [21]. Its basic idea is using the over-complete dictionary and sparse representation coefficient to restore the damaged image patch. Aharon *et al.* [22] proposed a K-SVD algorithm and used it to fill the missing pixels in images. Based on the method of Morphological Component Analysis (MCA), Elad *et al.* [23] proposed an inpainting method which can simultaneously restore the overlapping texture layer and cartoon layer. For smooth images, these methods can obtain good visual results. However, these methods also suffer from some limitations. For example, it takes a lot of time to learn a dictionary. Besides, the over-complete dictionary has a very important influence on the restoration effect, if the dictionary lacks of good adaptability, some texture details may be lost and the restored images are over-smooth.

In recent years, with the rapid development of deep learning technology, researchers began to introduce deep learning into image inpainting and proposed some inpainting methods based on deep network model. The basic idea of these methods is to use a large number of real images to train the generative model and the discriminative model, so that the deep network can learn the feature distribution of the real images, and then use the generative model to automatically generate the image of the damaged region to achieve the purpose of image restoration [24]. Zeng *et al.* [25] proposed a Pyramid-context ENcoder Network (PEN-Net) for image inpainting, they filled the damaged region by attention transfer from deep to shallow in a pyramid fashion, and obtained more realistic results. Sagong *et al.* [26] proposed the PEPSI (Parallel Extended-decoder Path for Semantic Inpainting) model. They used a structure consisting of a single shared encoding network and a parallel decoding network to reduce the number of convolution operations, and obtained superior performance to other models in terms of testing time. Jiang *et al.* [27] used a generator, a global discriminator, and a local discriminator to design the network model, and generated more realistic restoration results.

From the perspective of image inpainting, removing object from image means that we need to restore the large missing region [28]. Recently, the most commonly used method is proposed by Criminisi *et al.* [17]. However, it still suffers some problems. For example, it uses the SSD (Sum of Squared Differences) to measure the degree of similarity between exemplar patch and target patch. Although the matching rule is simple, it may result in that the target patch is filled by the inappropriate exemplar patch, which can lead to the mismatch error. Even worse, due to the irreversibility of the restoration process, the error may be accumulated along with the process progresses. At last, some undesired objects may be introduced into the restored images, making the results unable to meet the requirements of visual consistency. In view of these problems, we propose an inpainting method based on adaptive two-round search strategy. Compared with other methods, our main contributions are as follows:

1. In order to effectively measure the difference between exemplar patch and target patch, we define the DBP (Differences Between Patches) between the two patches, and use it to measure the degree of difference between the pixels already exist and the pixels to be filled.
2. In order to timely judge the occurrence of mismatch error, based on SSD and DBP, we continuously monitor the restoration process and adaptively determine whether a mismatch error between the target patch and the exemplar patch has occurred.
3. In order to timely prevent the mismatch error and error accumulation, we define a new matching rule and implement a two-round search strategy. We re-search the most similar exemplar patch in the source region according to the rule and use the new exemplar patch to restore the target patch.

The rest of this paper is organized as follows: In Section 2, we introduce some related works and the basic idea of proposed method. In Section 3, the details of the proposed method are described. We analyze causes of the mismatch error, and define the DBP between the exemplar patch and the target patch to measure the degree of difference of these two patches. Besides, we define a new matching rule and implement a two-round search strategy. In addition, we describe our algorithm steps in detail. The experiments and comparisons are performed in the Section 4. Finally, we conclude this work in Section 5.

II. RELATED WORKS

Aiming at the problems existing in the traditional methods for object removal, researchers have put forward a variety of methods [29]. Shen *et al.* [30] directly selected some undamaged patches from original image, formed an over-complete dictionary, and restored the damaged images based on the sparse representation. For smooth images, the method can obtain satisfactory restoration effect. But for texture images, it may lead to the loss of some texture details. In the method proposed by Liu *et al.* [31], it modified the confidence term into an exponential form and computed the sum of confidence term and data term to make the filling order more reasonable. In the method proposed by Liu *et al.* [32], the comparison of the variance between damaged patch and exemplar patch was added to the matching rule. Florinalbel *et al.* [33] added the edge information to the matching rule, in order to find the most matched exemplar patch. Choi and Hahm [34] set the search region according to the local feature information, so as to reduce the mismatch error. Xu and Sun [35] proposed the notation of patch structure sparsity, and proposed a method based on patch propagation. The method can make the filling order more reasonable. However, it took a lot of time to calculate the similarity, which can affect the efficiency of restoration. Zhang *et al.* [36] used the information of curvature and gradient to replace the data term, to improve the filling order. Wong and Orchard [37] introduced the idea of non-local means used in image denoising to image inpainting. They used the mean of several exemplar patches instead of a single exemplar patch to restore target patch. It can timely reduce the mismatch error and improve the restoration effect. However, it used the mean of a number of exemplar patches to restore each target patch, which will result in that some texture details in images are lost, and leads to the over-smooth phenomenon in the target region. Nan and Xi [38] set different weights for data item and confidence item according to the golden section, which make the restoration order more reasonable, but it cannot effectively prevent the occurrence of mismatch error, and the restoration effect needs to be improved. Isogawa *et al.* [39] proposed an approach to optimize the shape of masked. The advantage of this approach does not depend on inpainting algorithms, thus it can be applied for every inpainting method.

In this paper, we define the DBP between the exemplar patch and the target patch, and use the SSD and DBP to adaptively detect the occurrence of mismatch error. Besides, if there is a mismatch error, we define a new matching rule and implement a two-round search strategy to re-search the exemplar patch. In this way, we can effectively prevent the occurrence of mismatch error and error accumulation, and make the restoration image meet the requirements of human vision.

III. PROPOSED METHOD

A. NOTATIONS

From the perspective of object removal, the essence of image inpainting is to use the effective information in the image to fill the target region where the object is located. For easy understanding, we adopt same notations used in [17], as shown in Fig. 1, Ω is the target region (i.e., the missing region) which will be removed and filled, Φ is the source region (i.e., the known region), it may be defined as the entire image I minus the target region Ω ($\Phi = I - \Omega$), $\partial\Omega$ denotes the boundary of the target region Ω. Suppose that the patch Ψ_p centered at the point $\mathrm{p}(\mathrm{p} \in \partial\Omega)$ is to be filled. Given the patch Ψ_p, $\mathrm{n_p}$ is the unit vector orthogonal to the boundary $\partial\Omega$ and $\nabla I_\mathrm{p}^\perp$ is the isophote at point p.

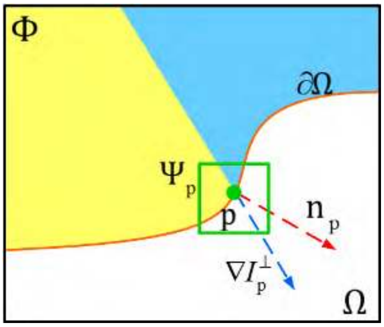


FIGURE 1. Notation diagram.

B. PRIORITY COMPUTATION

Filling order is crucial and depends entirely on the priority values that are assigned to each patch. Here we follow [17] to compute the patch priority and determine the filling order, because it is biased toward those patches which are on the continuation of strong edges and which are surrounded by high-confidence pixels, it can reserve the structure information efficiently.

For each patch Ψ_p centered at the point $\mathrm{p}(\mathrm{p} \in \partial\Omega)$ has a patch priority, it is defined as:

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

where $C(\mathrm{p})$ is the confidence term and $D(\mathrm{p})$ is the data term. The confidence term $C(\mathrm{p})$ indicates how many existing pixels are there in the target patch. The more pixels already exist, the higher the confidence is. It is defined as:

$$C(\mathrm{p}) = \frac{\sum_{\mathrm{q}\in\Psi_\mathrm{p}\cap\bar{\Omega}} C(\mathrm{q})}{\left|\Psi_\mathrm{p}\right|} \tag{2}$$

where $\left|\Psi_\mathrm{p}\right|$ is the area of patch, i.e., the number of pixels in Ψ_p. During the initialization, $C(\mathrm{p})$ is set as:

$$C(\mathrm{p}) = \begin{cases} 0 & \forall \mathrm{p} \in \Omega \\ 1 & \forall \mathrm{p} \in \Phi \end{cases} \tag{3}$$

The data term $D(\mathrm{p})$ indicates how strong the isophote hitting the boundary is. It is especially important because it encourages the linear structure to be synthesized first. It is defined as:

$$D(\mathrm{p}) = \frac{\left|\nabla I_\mathrm{p}^\perp \cdot \mathrm{n_p}\right|}{\alpha} \tag{4}$$

where α is a normalization factor, for a typical grey-level image, its value is 255. Once all priorities have been computed, we find the target patch Ψ_p with the highest priority.

After the target patch is determined, we search in the source region for the exemplar patch which is most similar to target patch according to the matching rule.

C. DEFINATION OF DBP

In the traditional exemplar-based inpainting method, the SSD is used to measure the degree of similarity between target

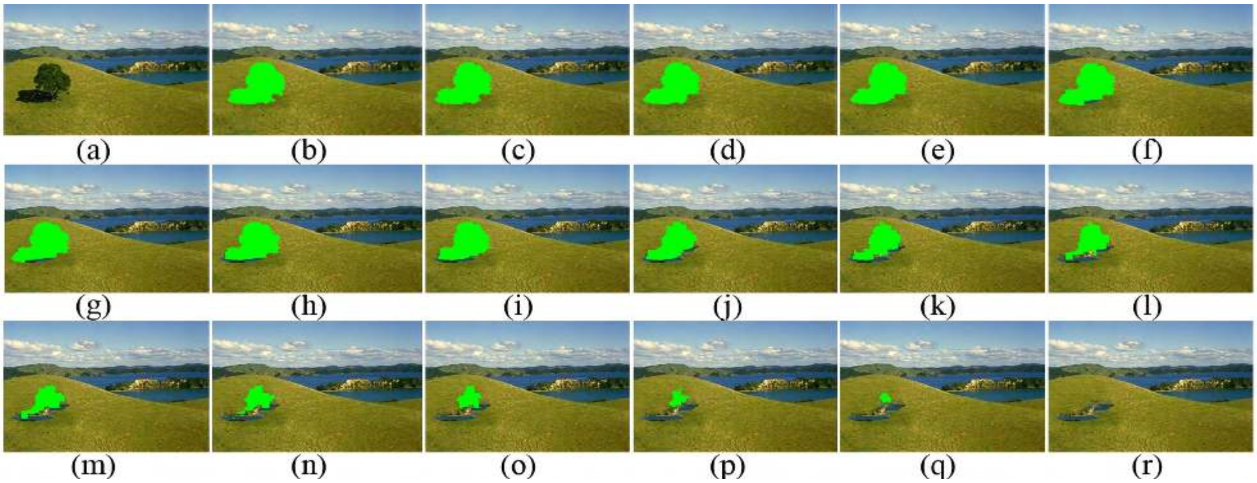


FIGURE 2. Restoration process of the method in [17].

patch and exemplar patch. It is defined as follows [17]:

$$ssd(\Psi_p, \Psi_q) = \sum \left|\bar{M}\Psi_p - \bar{M}\Psi_q\right|^2 \tag{5}$$

where Ψ_p is the target patch and Ψ_q is the exemplar patch, M is the binary mask, it uses one to indicate the pixels that need to be filled, and uses zero to indicate the already existing pixels.

Based on the SSD, the matching rule is defined as follows:

$$\Psi_q = \underset{\Psi_{\hat{q}} \in \Phi}{\arg\min}\; ssd(\Psi_p, \Psi_{\hat{q}}) \tag{6}$$

The matching rule in Eq. (6) is simple, but it may cause the target patch to be replaced by an inappropriate exemplar patch. Even worse, the mismatch error will be continually accumulated along with the process progresses. Finally some unexpected and undesired objects may be introduced into target region, and the restored images cannot meet the requirements of human vision. For better illustration, we show an example in Fig. 2.

In Fig. 2, (a) is an original image which is obtained from the BSDS dataset [40], (b) is the object to be removed, which is marked by green, and (c)-(q) are the restoration process of method in [17] when 4th step, 7th step, 10th step, 15th step, 20thstep, 30th step, 40th step, 60th step, 80th step, 100th step, 120th step, 140th step, 160th step, 180th step, and 200th step, respectively. (r) is the final restoration result.

As can be seen from Fig. 2, the target region is decreasing along with the restoration process progresses in (c)-(f). However, there is a small blue object occurs in the target region in (g), which means that a small part of the blue lake is copied into the target region, and a mismatch error occurs between the target patch and the exemplar patch. In (h)-(q), the undesired blue object is increasing along with the restoration process continues, which means that the error is continuously accumulated. Finally, some unexpected objects are introduced into the target region, as shown in (r), the result cannot meet the requirements of human vision consistency.

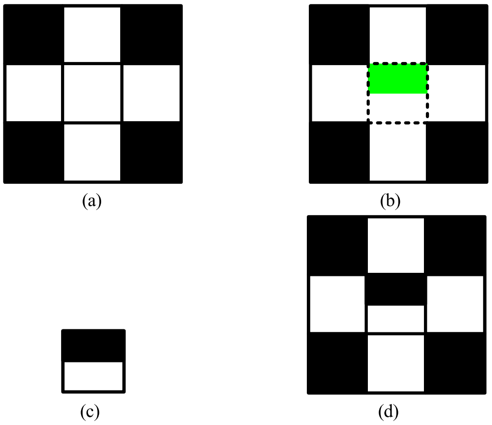


FIGURE 3. An example of mismatch error.

By analyzing the reasons for the above situation, we find that although the value of SSD is very small (i.e., the existing pixels in target patch and the corresponding pixels in exemplar patch are similar), there may be a mismatch error between them. In fact, even if the value of SSD is 0, the mismatch error may occur. For better illustration, we synthesize an image in Fig. 3, where (a) is the original image, which contains nine small squares, and these small squares are the same size. Suppose that the patch size is equal to the small square. The dotted line in (b) is the target patch, and the target region is marked by green, i.e., the upper part of the target patch is to be restored. According to the matching rule in Eq. (2), (c) is the exemplar patch because the value of SSD is 0. (d) is the result of using (c) to restore the target region. From (d) we can see that it has obvious visual inconsistencies in the target patch.

For better explanation, the reason for mismatch error in Fig. 3 is analyzed in Fig. 4, where the first is target patch, A is the damaged region to be filled, B is the undamaged region. The second is exemplar patch, C is the region corresponding to A, and D is the region corresponding to B. The

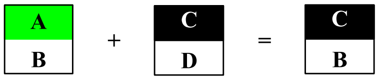


FIGURE 4. Analysis of mismatch in Figure 3.

third is the restored patch. Both B and D are white, so the value of SSD is 0. However, C is black while B is white, i.e., the difference between the pixels used to fill and the pixels already exist is large. Therefore there is an obvious visual inconsistency in the target patch, and the mismatch error occurs.

Based on the above analysis, we know that the situations under which mismatch error is likely to occur can be summarized into two categories. First, if the already existing pixels in target patch and the corresponding pixels in exemplar patch are quite different, the mismatch error is likely to occur. The situation can be judged according to Eq. (5). If the value of SSD is relatively large, it means there is a potential mismatch error between the two patches. In this case, if the target patch is replaced by the exemplar patch, it is likely to result in a mismatch error.

Second, if the difference between the pixels used to fill in exemplar patch and the pixels already exist in target patch is relatively large, the mismatch error is likely to occur. In this situation, when using the exemplar patch to restore the target patch, there is a great difference between the two parts of the restored patch, which may lead to obvious visual inconsistency, as shown in Fig. 3(d). In view of this situation, we define the DBP between the two patches as follows:

$$dbp(\Psi_{\mathrm{p}}, \Psi_{\mathrm{q}}) = \left| \frac{\sum \bar{M}\Psi_{\mathrm{p}}}{\sum \bar{M}} - \frac{\sum M\Psi_{\mathrm{q}}}{\sum M} \right|^2 \tag{7}$$

where the first item calculates the mean value of pixels already exist in the target patch, the second item calculates the mean value of pixels used to fill in the exemplar patch.

In short, we use both SSD and DBP to measure the degree of difference between the exemplar patch and the target patch. Specifically, we use the SSD to measure the degree of difference between existing pixels, and use the DBP to measure the degree of difference between the pixels already exist and the pixels to be filled. If the value of SSD is relatively large or the value of DBP is relatively large, we think there is a mismatch error between the two patches. In this way, we can effectively judge the occurrence of mismatch error.

D. ADAPTIVE TWO-ROUND SEARCH STRATEGY

During the restoration process, we monitor each pair of target patch and exemplar patch, and adaptively judge whether there is a mismatch between them. If there is a mismatch error, a two-round search strategy is implemented.

In order to timely prevent the occurrence of mismatch error and error accumulation, we use adaptive thresholds to judge the mismatch error between target patch and exemplar patch. We set two thresholds β and γ, when the value of SSD is larger than β or the value of DBP is larger than γ, we think the mismatch error between exemplar patch and target patch occurs. In this situation, we define the new matching rule and implement the two-round search strategy to find the exemplar patch.

Algorithm 1 Proposed Algorithm

```
Input: Original image $I$ and corresponding mask $M$
Output: Restored image $I_r$
while $M$ do
    for each pixel p in $\partial\Omega$ do
        $P(\mathrm{p}) = C(\mathrm{p}) \times D(\mathrm{p})$
    end for
    $\Psi_{\mathrm{p}} = \max(P(\mathrm{p}))$
    for each patch $\Psi_{\hat{q}}$ in $\Phi$ do
        $ssd(\Psi_{\mathrm{p}}, \Psi_{\hat{q}}) = \sum \left|\bar{M}\Psi_{\mathrm{p}} - \bar{M}\Psi_{\hat{q}}\right|^2$
    end for
    $\Psi_{\mathrm{q}} = \min(\, ssd(\Psi_{\mathrm{p}}, \Psi_{\hat{q}}))$
    $dbp(\Psi_{\mathrm{p}}, \Psi_{\mathrm{q}}) = \left|\sum \bar{M}\Psi_{\mathrm{p}} / \sum \bar{M} - \sum M\Psi_{\mathrm{q}} / \sum M\right|^2$
    if $ssd(\Psi_{\mathrm{p}}, \Psi_{\mathrm{q}}) > \beta$ or $dbp(\Psi_{\mathrm{p}}, \Psi_{\mathrm{q}}) > \gamma$ then
        for each patch $\Psi_{\hat{q}}$in $\Phi$ do
            $\Psi_{\mathrm{q}'} = ssd(\Psi_{\mathrm{p}}, \Psi_{\hat{q}}) + \omega \times dbp(\Psi_{\mathrm{p}}, \Psi_{\hat{q}})$
        end for
        $\Psi_{\mathrm{q}} = \min(\, \Psi_{\mathrm{q}'})$
    end if
    $M\Psi_{\mathrm{p}} = M\Psi_{\mathrm{q}}$
end while
```

In the proposed method, we adopt a very simple and effective method to determine the thresholds. First we calculate the SSD and DBP of all the matched patches and obtain two arrays: *ssd_data* and *dbp_data*. Then, we sort the two arrays in ascending order, and get the two sorted arrays: *ssd_data_sort* and *dbp_data_sort*. At last, we can obtain the thresholds β and γ as follows:

$$\begin{cases} \beta = ssd\_data\_sort[\lambda_1 * num] \\ \gamma = dbp\_data\_sort[\lambda_2 * num] \end{cases} \tag{8}$$

where λ_1 and λ_2 are the proportional coefficients, respectively. *num* is the number of matched patches.

Through analysis, we find that the number of patches with mismatches accounts for a small proportion. For example, we calculate the SSD and DBP of all the matched patches in Fig. 2 and show them in Fig. 5 and Fig. 6, respectively. Fig.5 is the distribution of SSD, and Fig. 6 is the distribution of DBP. We can see that the vast majority of the data distributions are relatively concentrated, and their values are relatively small, that is, there is no mismatch in the vast majority of matched patches. In comparison, only a small number of data are scattered, and their values are relatively large, that is, there is mismatch in a small number of matched patches. Therefore, the values of λ_1 and λ_2 in the Eq. (8) are generally between 0.7 and 0.9.

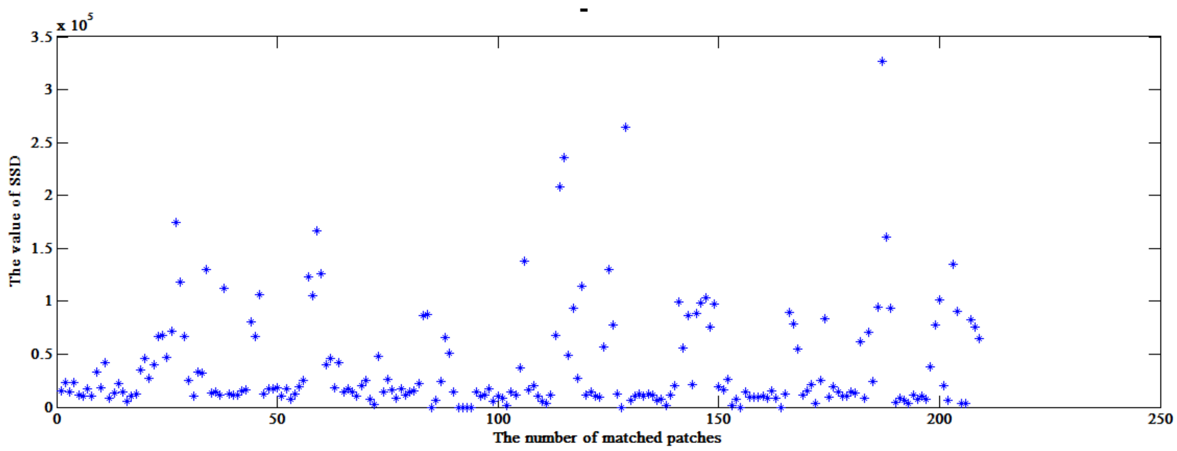


FIGURE 5. The distribution of SSD.

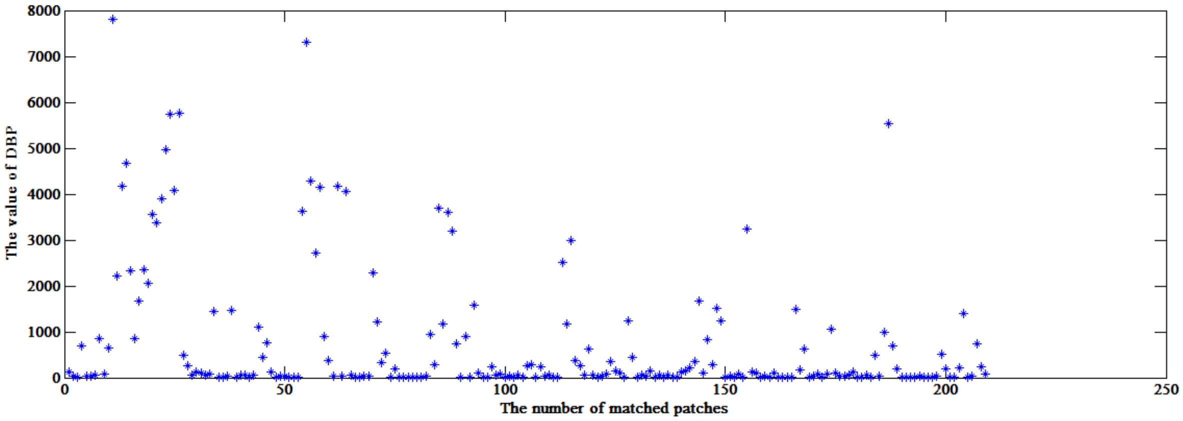


FIGURE 6. The distribution of DBP.

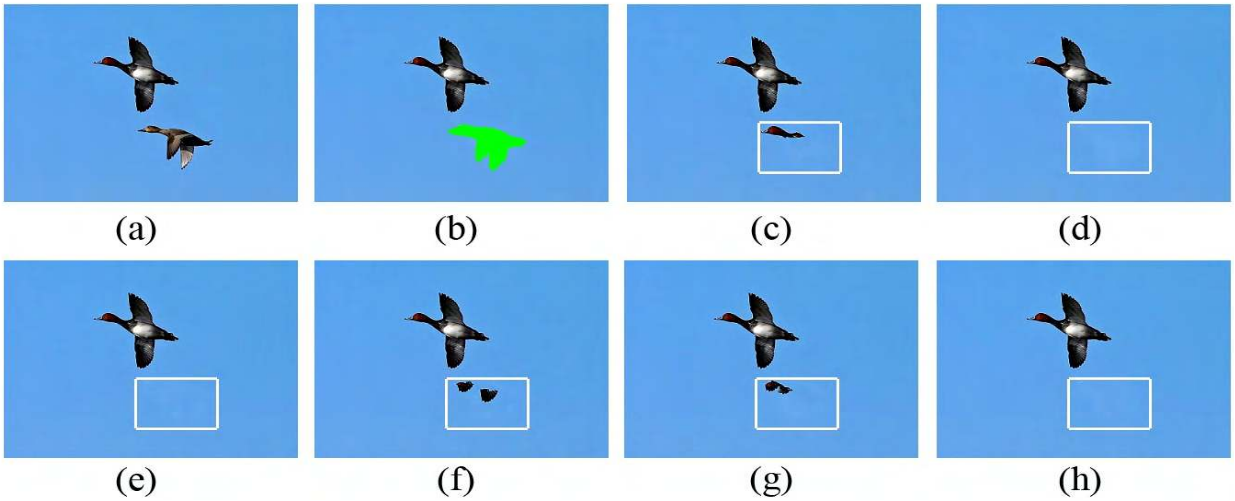


FIGURE 7. Restoration results of "goose". (a) is the original image, (b) is the target region, which is marked by green, (c) is the result of method in [17], (d) is the result of method in [30], (e) is the result of method in [31], (f) is the result of method in [36], (g) is the result of method in [38], and (h) is the result of proposed method. For better comparisons, in the restoration results of each method, we marked the target region with a white rectangle.

Based on the thresholds β and γ, we can adaptively judge the occurrence of mismatch error. If there is a mismatch error between the exemplar patch and target patch, the two-round search strategy is adaptively implemented.

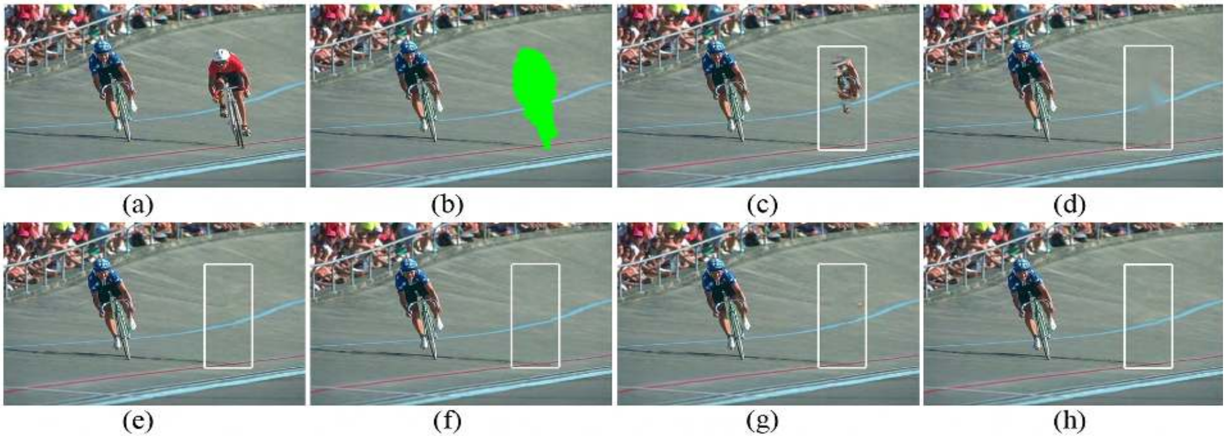


FIGURE 8. Restoration results of "cyclist". (a) is the original image, (b) is the target region, which is marked by green, (c) is the result of method in [17], (d) is the result of method in [30], (e) is the result of method in [31], (f) is the result of method in [36], (g) is the result of method in [38], and (h) is the result of proposed method. For better comparisons, in the restoration results of each method, we marked the target region with a white rectangle.

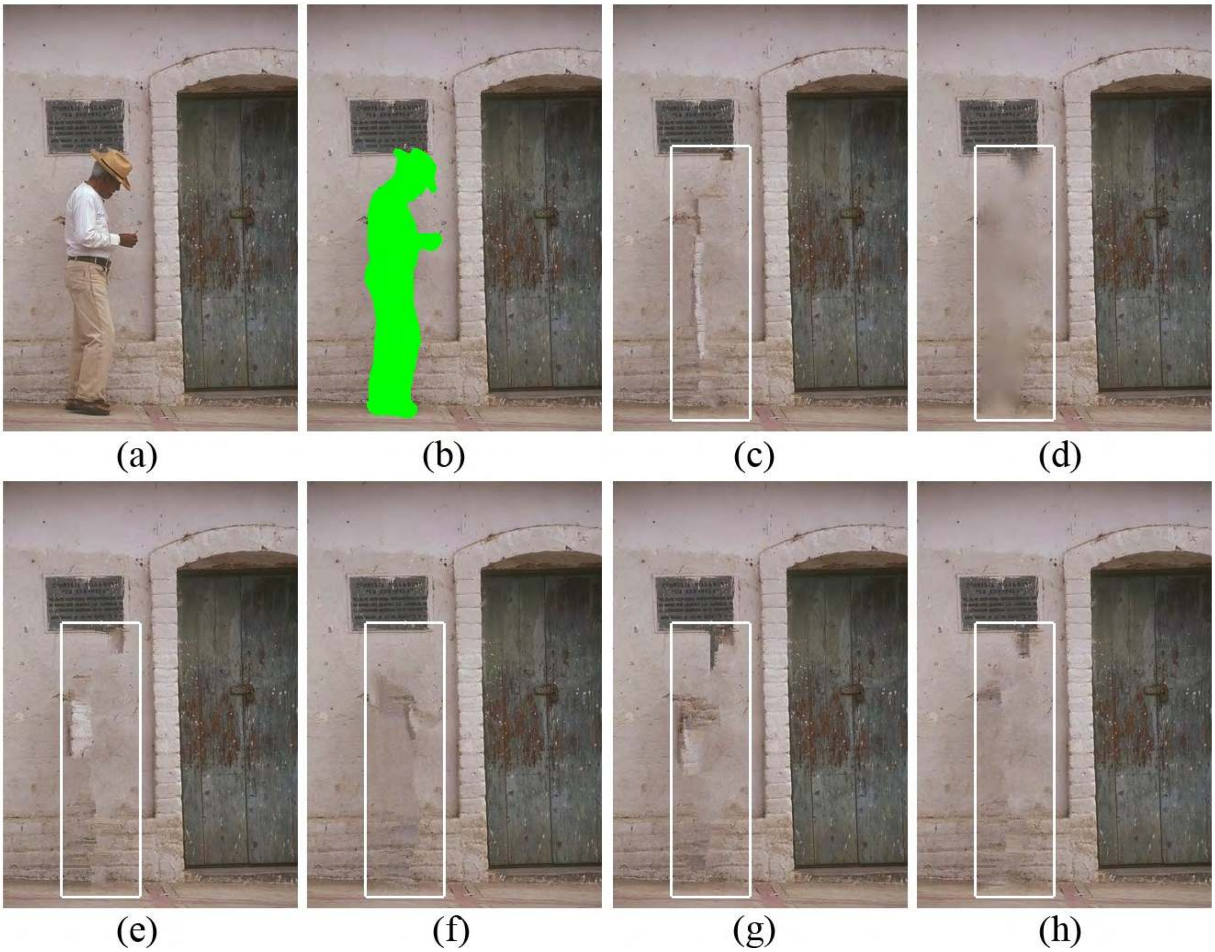


FIGURE 9. Restoration results of "old man". (a) is the original image, (b) is the target region, which is marked by green, (c) is the result of method in [17], (d) is the result of method in [30], (e) is the result of method in [31], (f) is the result of method in [36], (g) is the result of method in [38], and (h) is the result of proposed method. For better comparisons, in the restoration results of each method, we marked the target region with a white rectangle.

In order to re-research the most similar exemplar patch, we redefine a new matching rule in the two-round search strategy. We know the reason of the mismatch error is that the SSD between the target patch and the exemplar patch is relatively large, or the DBP between the two patches is relatively large. Therefore, we define the new matching rule

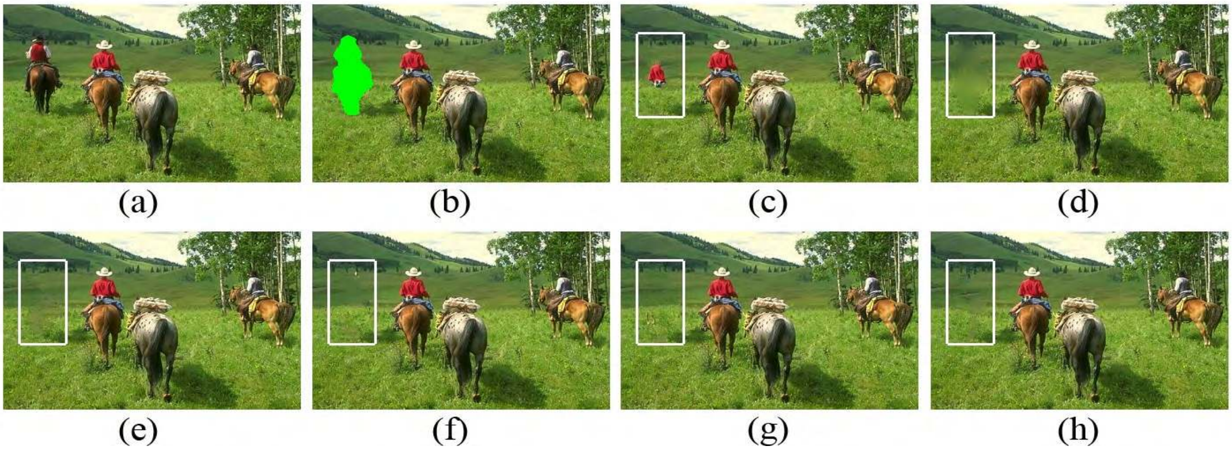


FIGURE 10. Restoration results of "rider". (a) is the original image, (b) is the target region, which is marked by green, (c) is the result of method in [17], (d) is the result of method in [30], (e) is the result of method in [31], (f) is the result of method in [36], (g) is the result of method in [38], and (h) is the result of proposed method. For better comparisons, in the restoration results of each method, we marked the target region with a white rectangle.

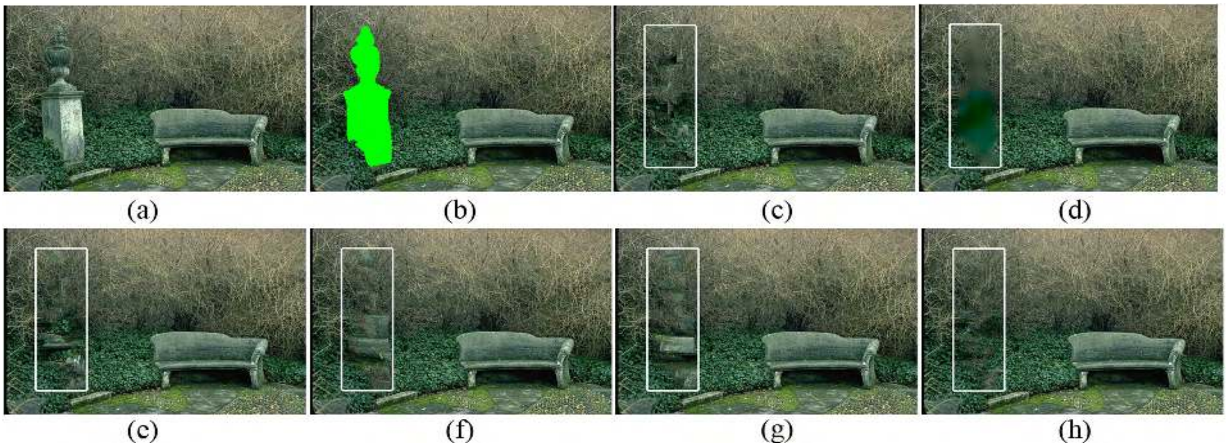


FIGURE 11. Restoration results of "stone column". (a) is the original image, (b) is the target region, which is marked by green, (c) is the result of method in [17], (d) is the result of method in [30], (e) is the result of method in [31], (f) is the result of method in [36], (g) is the result of method in [38], and (h) is the result of proposed method. For better comparisons, in the restoration results of each method, we marked the target region with a white rectangle.

using SSD and DBP as follows:

$$\Psi_{q'} = \mathop{\arg\min}_{\Psi_{\hat{q}} \in \Phi} ssd(\Psi_p, \Psi_{\hat{q}}) + \omega \times dbp(\Psi_p, \Psi_{\hat{q}}) \tag{9}$$

where ω is an adjustment factor that makes the SSD and the DBP equal at the order of magnitude. It is defined as follows:

$$\omega = \sum \bar{M} \tag{10}$$

As can be seen from Eq. (9), when the two-round search strategy is implemented, we not only measure the degree of difference between the already existing pixels, but also measure the degree of difference between the pixels already exist and the pixels to be filled. In this way, we can re-search the most similar exemplar patch, and timely prevent the occurrence of mismatch error.

E. PATCH RESTORATION

When the most matching exemplar patch is found, we calculate the SSD and DBP between target patch and exemplar patch, compare them with their respective threshold, and adaptively select different methods to restore the target patch according to different situations.

If the value of SSD is larger than β or the value of DBP is larger than γ, we think the mismatch error occurs. Then the two-round search strategy is implemented, we re-search the exemplar patches in the source region according to Eq. (9). Finally, the target patch can be restored as follows:

$$M\Psi_p = M\Psi_{q'} \tag{11}$$

where $\Psi_{q'}$ is the exemplar patch found according to Eq. (9).

Otherwise we think that the current exemplar patch is the most similar exemplar patch, and no mismatch error occurs. The target patch can be restored by the exemplar patch as

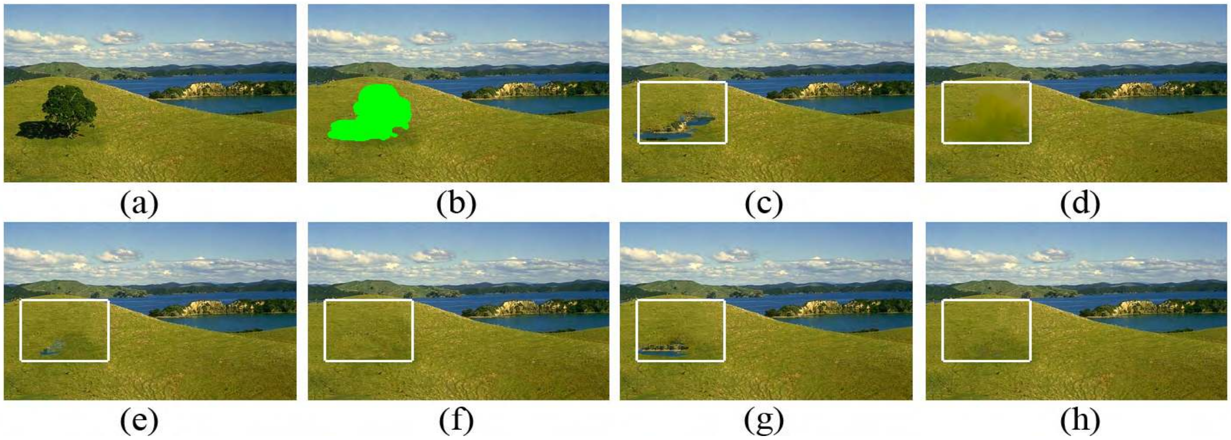
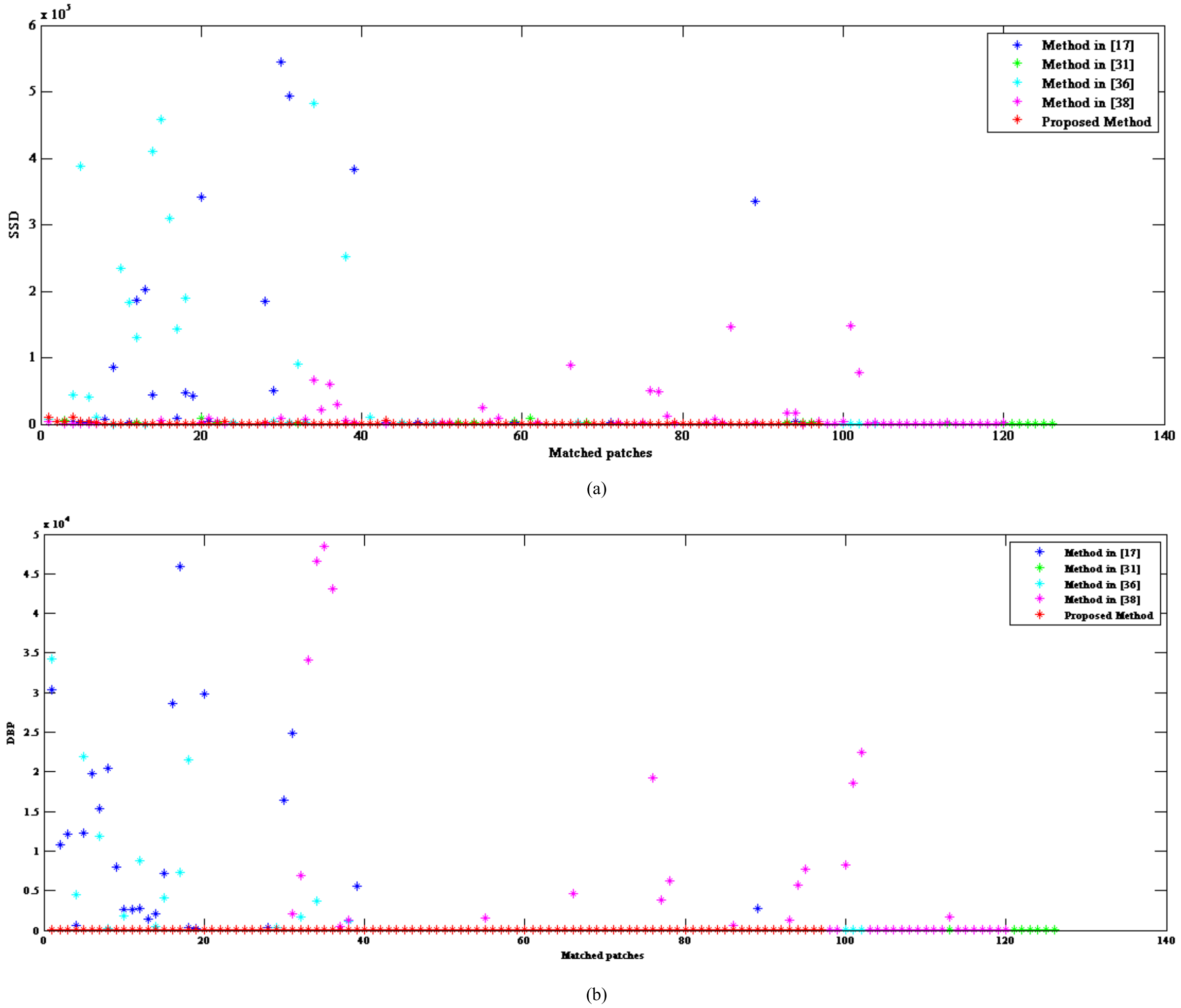


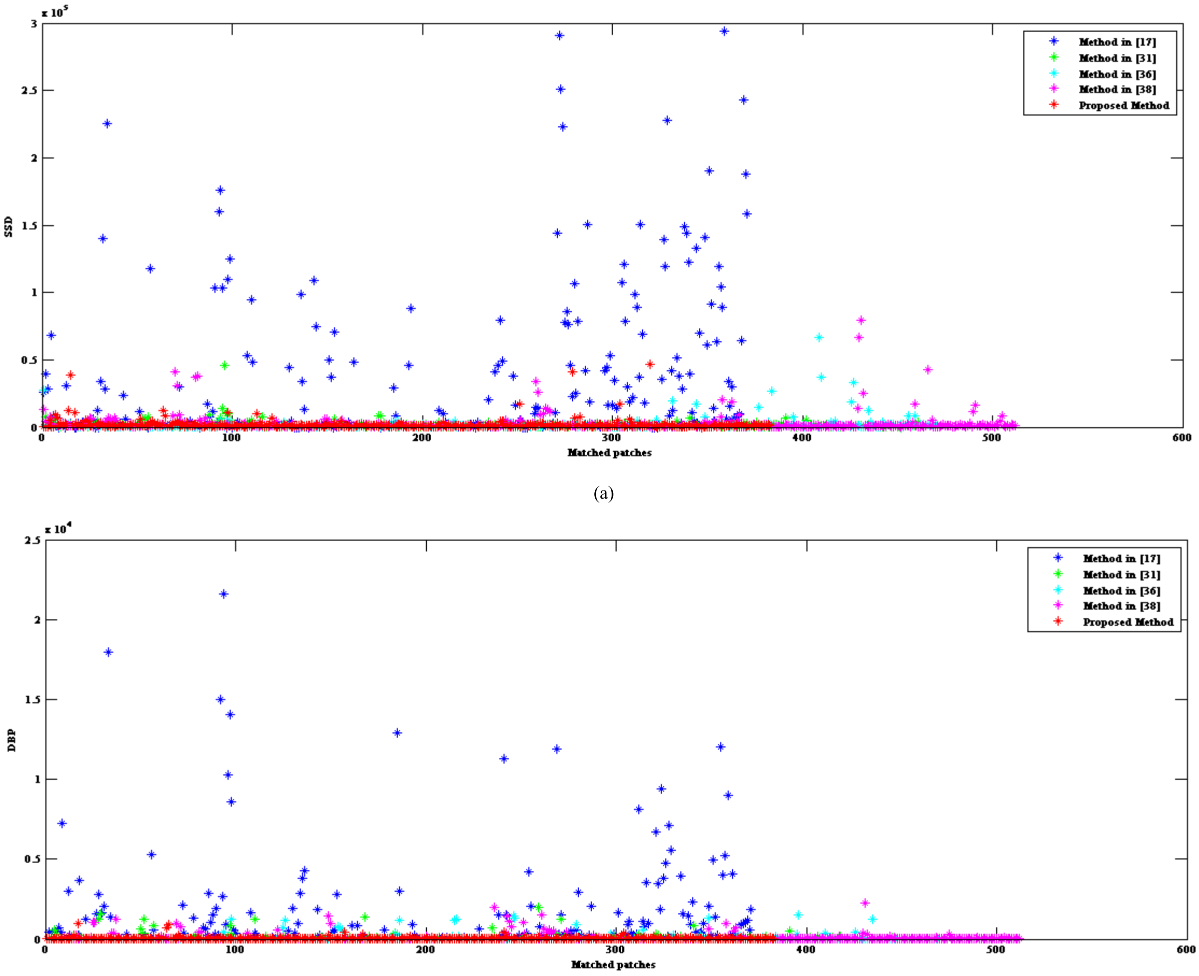
FIGURE 12. Restoration results of "hillside". (a) is the original image, (b) is the target region, which is marked by green, (c) is the result of method in [17], (d) is the result of method in [30], (e) is the result of method in [31], (f) is the result of method in [36], (g) is the result of method in [38], and (h) is the result of proposed method. For better comparisons, in the restoration results of each method, we marked the target region with a white rectangle.



(a)

(b)

FIGURE 13. The SSD and DBP of each method for "goose". (a) is the SSD distribution of these methods, and (b) is the DBP distribution of these methods.



(a)

(b)

FIGURE 14. The SSD and DBP of each method for "cyclist". (a) is the SSD distribution of these methods, and (b) is the DBP distribution of these methods.

follows:

$$M\Psi_{\mathrm{p}} = M\Psi_{\mathrm{q}} \tag{12}$$

where Ψ_{q} is the exemplar patch found according to Eq. (6).

F. ALGORITHM DESCRIPTION

Here, the proposed algorithm is described as follows:

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In order to verify the feasibility and effectiveness of the proposed method, we select a variety of natural images to experiment. All the experiments are run on the computer with the configuration of 3.7GHz processor and 4GB RAM.

For each image, we specify a target object, and then restore the region where it is located, to achieve the goal of removing the object from the image. For better comparison and analysis, in the experiment, we use the method in [17], the method in [30], the method in [31], the method in [36], the method in [38], and the proposed method to restore the images, respectively, and compare and analyze their restoration results.

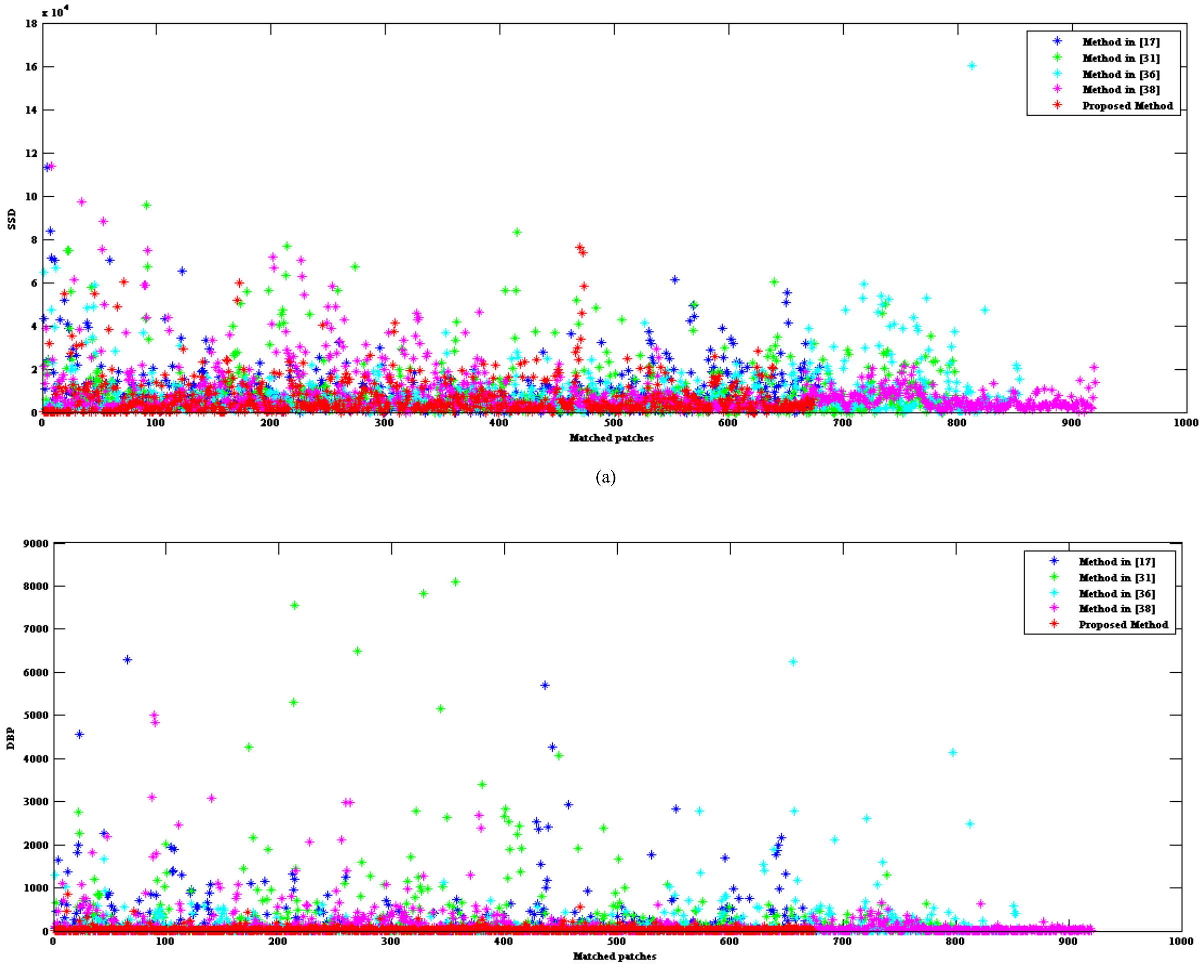
To better illustrate, experimental results are divided into two groups to demonstrate and analyze. The first group is the results of smooth images, and the second group is the results of texture images. All the original images are obtained from the BSDS dataset [40].

A. RESULTS OF SMOOTH IMAGES

Here we demonstrate the restoration results of three smooth images in Figs. 7- 9. In Fig. 7, we remove a goose flying in the sky from the image, so we named the image as "goose". In Fig. 8, we remove the right cyclist from the image, so we named the image as "cyclist". In Fig. 9, we remove the old man from the image, so we named the image as "old man".

B. RESULTS OF TEXTURE IMAGES

Here we demonstrate the restoration results of three texture images in Figs. 10-12. In Fig. 10, we remove the rider on



(a)

(b)

FIGURE 15. The SSD and DBP of each method for "old man". (a) is the SSD distribution of these methods, and (b) is the DBP distribution of these methods.

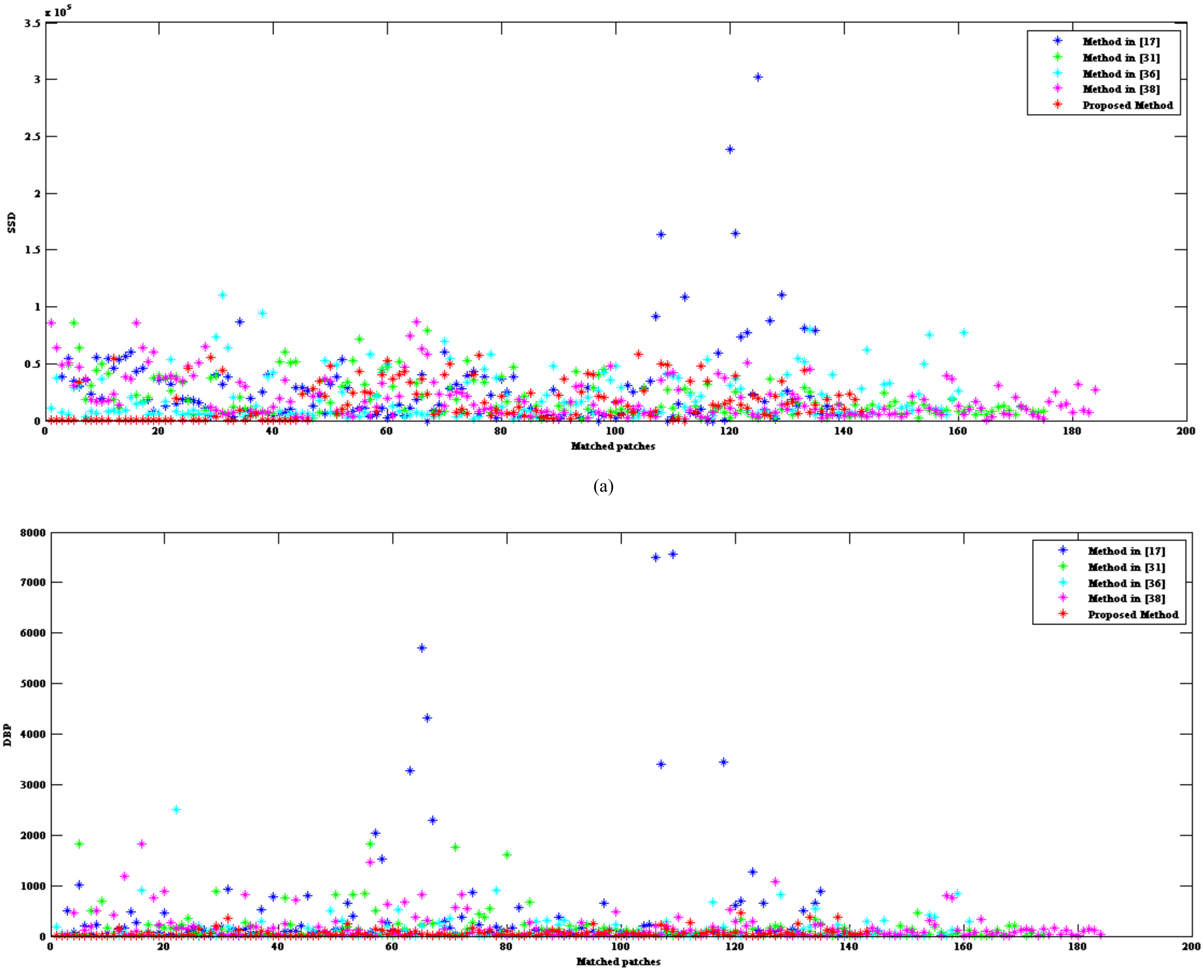
the left from the image, so we named the image as "rider". In Fig. 11, we remove the stone column from the image, so we named the image as "stone column". In Fig. 12, we remove the tree on the hillside from the image, so we named the image as "hillside".

C. COMPARISON AND ANALYSIS

As can be seen from above figures, in the results of method in [17], there are some undesired and unexpected objects in the target regions, which make the restored images cannot meet the visual consistency requirements. For example, in Fig. 7 (c), the head of another goose is copied into the target region. In Fig. 8 (c), some parts of another cyclist have occurred in the target region. In Fig. 9 (c), parts of the wall are copied into the target region, and a distinct irregular linear object appears in the image. In Fig. 10 (c), parts of another person's red coat have appeared in the target region. In Fig. 11 (c), parts of the stone bench have been copied into the target region. In Fig. 12 (c), parts of blue lake have occurred in the target region. The reason is that the mismatch error between target patch and exemplar patch occurs during the restoration process, and the error has been continuously accumulated along with the process continues, which cause some undesired objects to be introduced into the image.

The method in [30] used the sparse representation to restore the target patch. It can obtain satisfactory results in very smooth images, as shown in Fig. 7 (d). However, if the image contains a small count of edge or texture structure, it can lead to obvious over-smooth phenomenon and lose the details of the image, as shown in Fig. 8 (d) and Fig. 9 (d). For the images which contain rich texture, as shown in Figs. 10 (d), 11 (d), and 12 (d), although there are no unexpected objects in the target regions, a lot of texture details are lost, causing the images to be over-smooth.

The method in [31] modified the confidence term into an exponential form and computed the sum of confidence term and data term, which can improve the restoration effect to a certain extent. For example in Figs. 7 (e), 8 (e), and 10 (e),



(a)

(b)

FIGURE 16. The SSD and DBP of each method for "rider". (a) is the SSD distribution of these methods, and (b) is the DBP distribution of these methods.

it can obtain satisfactory results and we can hardly notice the restoration traces. In Fig. 12 (e), the size of the blue lake is significantly reduced. But it cannot effectively prevent the occurrence of mismatch and error accumulation, as shown in Fig. 9 (e), parts of the wall are copied into the target region. In Fig. 11 (e), parts of the stone bench have been copied into the target region.

The method in [36] used the information of curvature and gradient to replace the data term, which can improve the restoration effect to some extent. For example, in Figs. 8 (f), 10 (f), and 12 (f), there are no unexpected objects in the target region, and in Fig. 9 (f), the size of the undesired object is significantly reduced. However, it does not effectively prevent mismatch error and error accumulation. For example, in Fig. 7 (f) some parts of another goose are copied into the target region, and in Fig. 11 (f), parts of the stone bench appear in the target region.

The method in [38] set different weights for data item and confidence item according to the golden section, which can make the filling order more reasonable and improve the restoration effect. For example, there are no obvious unexpected objects in the target regions in Fig. 8 (g). In Fig. 10 (g), the undesired objects in the target regions are especially small and can be ignored if they are not carefully observed. However, the restoration effect need to be further improved. For example, some parts of another goose still appear the target region in Fig. 7 (g), parts of the wall are copied into the target region in Fig. 9 (g), and parts of the stone bench appear in the target region in Fig. 11 (g), parts of blue lake have occurred in the target region in Fig. 12 (g).

Compared with the other methods, the proposed method has obtained better results for each image. In Fig. 7 (h), the designated goose is completely removed, and no other objects are introduced into the target region. In Fig. 8 (h), the right cyclist is removed in an unnoticeable manner. In Fig. 9 (h), the old man is removed from the image. Although a small portion of the window is copied into the target region, it is much smaller than the objects introduced by other methods.

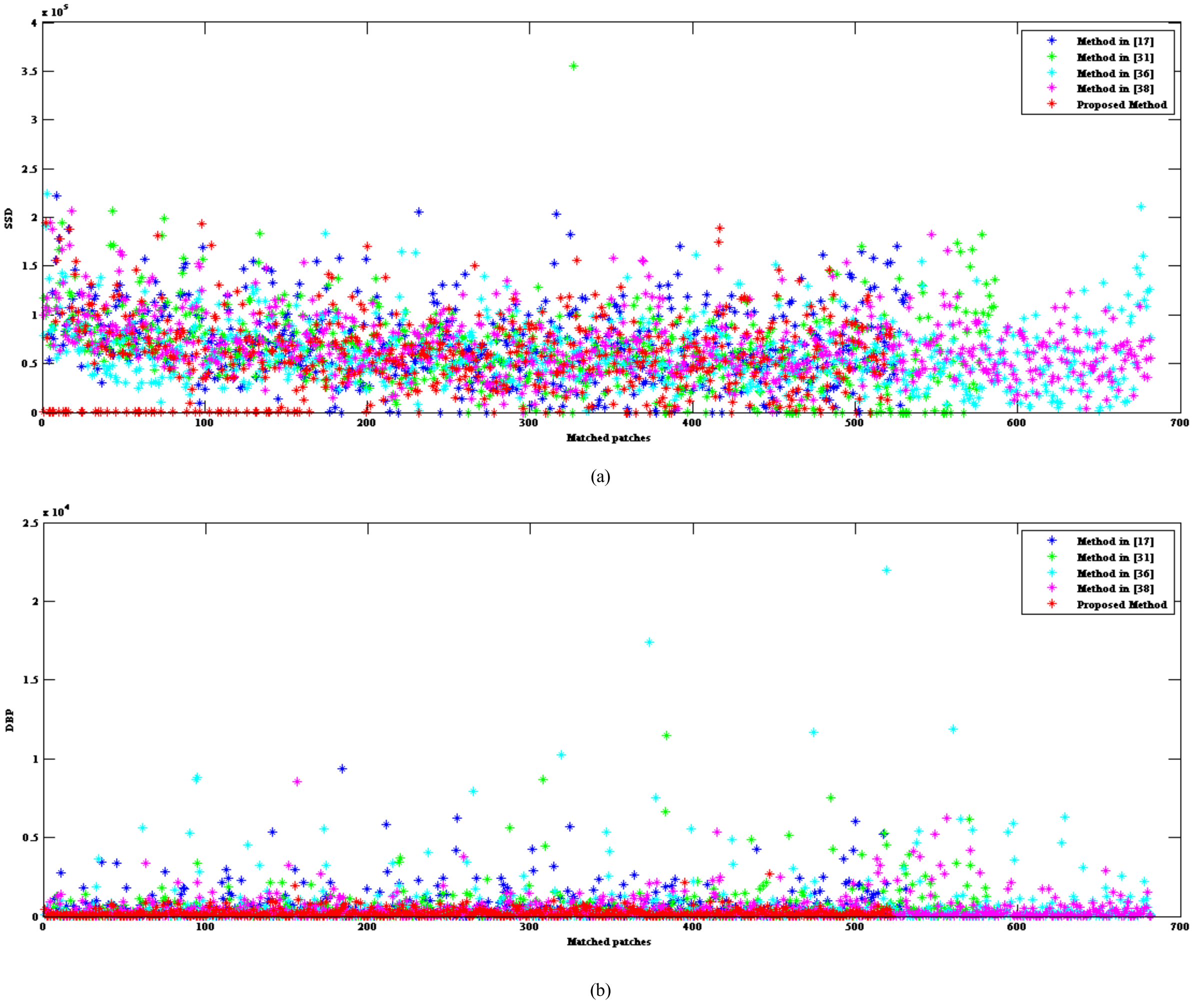


FIGURE 17. The SSD and DBP of each method for "stone column". (a) is the SSD distribution of these methods, and (b) is the DBP distribution of these methods.

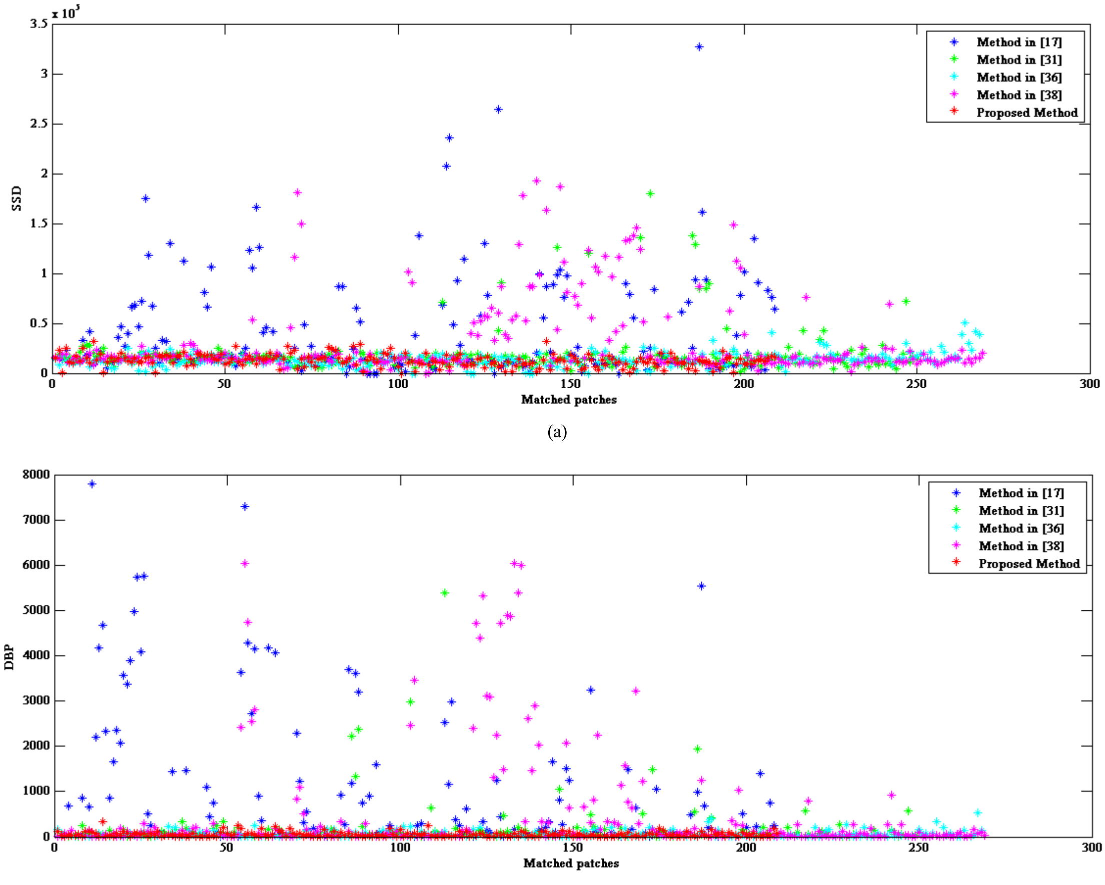
In Fig. 10 (h), the rider on the left is completely removed, and the target region is filled in a very natural way. In Fig. 11 (h), the stone column in the bush is removed and the target region is well restored. In Fig. 12 (h), the tree on the hillside is removed from the image, and we hardly notice the traces of restoration. The reason is that we continuously monitor the restoration process. If a mismatch error occurs, we perform a two-round search strategy based on the new matching rule defined. We re-research for the best similar exemplar patch and use it to replace the target patch. In this way, the proposed method can timely prevent the mismatch error and error accumulation, and make the results meet the requirements of visual consistency.

D. QUANTITATIVE COMPARISON

In proposed method, we use the SSD to measure the degree of similarity between existing pixels, and use the DBP to measure the degree of difference between the pixels already exist and the pixels to be filled. If the value of SSD is relatively large or the value of DBP is relatively large, we think there is a mismatch error between the two patches. In this case, we implement a two-round search strategy, and re-search the exemplar patch in source region. Therefore, compared with the traditional method, the SSD and DBP between exemplar patch and target patch are relatively small.

In order to verify whether the SSD and DBP of proposed method are relatively small, during the restoration process of each image, we respectively saved SSD and DBP of method in [17], [31], [36], [38], and proposed method. It should be noted that we did not save the values of method in [30]. The reason is that this method directly uses an over-complete dictionary and sparse coding to reconstruct the target patch without searching for exemplar patch in the source region. The SSD and DBP of each method for each image are shown in Figs 13-18.

As can be seen from the distribution of SSD and DBP in Figs 13-18, the SSD value and DBP value of our method are smaller than those of other methods. This means that our method can effectively avoid the large differences between the target patches and the exemplar patches, so as to prevent



(a)

(b)

FIGURE 18. The SSD and DBP of each method for "hillside". (a) is the SSD distribution of these methods, and (b) is the DBP distribution of these methods.

the target patch from being replaced by the inappropriate exemplar patch, resulting in the inconsistency of subjective vision. Also we can see that, if the image is smoother, the distribution of SSD and DBP is more concentrated, indicating that most of the exemplar patches are very similar to the target patches, as shown in Fig 13 and Fig 14. If the image contains more texture details, the distribution of SSD and DBP is more scattered, indicating that there are relatively large differences between the exemplar patches and the target patches, as shown in Fig 17 and Fig 18.

In order to show the effectiveness of our method more accurately from the specific data, we take the image "hillside" in Fig 12 as the representative and extract the five largest values of each method from Fig 18, as shown in Table 1 and Table 2. Table 1 shows the five largest SSD of each method, and Table 2 shows the five largest DBP of each method.

From the data distribution in above figures and tables, it can be seen that, the data of method in [17] is the most scattered, and includes the largest SSD value and DBP value of all methods. It means that a lot of mismatch errors occurred while the restoration process was going on, and a lot of unexpected objects were introduced into the target regions, which can be seen from the results shown in Fig. 12 (c).

TABLE 1. Five largest SSD of each method.

Method	SSD				
Method in [17]	326969	264207	236128	207669	174557
Method in [31]	179438	138069	136223	128561	125673
Method in [36]	50765	41245	40938	39206	38465
Method in [38]	192916	187061	180677	177673	162834
Proposed Method	31639	31574	28727	27327	27132

TABLE 2. Five largest DBP of each method.

Method	DBP				
Method in [17]	7800	7299	5762	5731	5542
Method in [31]	5382	2989	2373	2224	1933
Method in [36]	536	402	326	326	261
Method in [38]	6046	6040	6003	5396	5327
Proposed Method	345	242	241	231	213

The SSD and DBP of method in [31] are relatively smaller than that of methods in [17] and [38]. However, there are still some matched patches with large SSD and DBP. Therefore, we can see from Fig. 12 that, the size of unexpected object in (e) is smaller than that of (c) and (g), while there is a small undesired object in (e).

The data distribution of method in [36] is most similar to the proposed method, and both SSD and DBP are very small. Also, we can see from Fig. 12 (f), there is no undesired object in the target region, and the result can satisfy the requirements of human vision.

The values of SSD and DBP of method in [38] are only smaller than that of method in [17] and larger than that of other methods. Consistent with this situation, the size of the unexpected object in Fig. 12 (g) is only smaller than that in Fig. 12 (c), but larger than that of the other methods, and the restoration effect need to be greatly improved.

The values of our method are much smaller than that of other methods. It means that when a mismatch error occurs, our method re-searches the exemplar patch through the two-round search strategy, which can effectively prevent the occurrence of mismatch and the accumulation of errors, and prevent unexpected objects from appearing in the restored image, making the restoration image satisfy the requirements of human visual consistency, as shown in Fig. 12 (h).

E. DISCUSSION

Through the above qualitative and quantitative analysis and comparison, we can see that, compared with other methods, our method can effectively avoid the occurrence of mismatch error and obtain satisfactory restoration results. Also, compared with the method based on deep convolutional neural network, our method does not need to train a large number of samples, and does not need to spend a lot of time to continuously adjust the model parameters. Here we will also mention our method's limitation. In essence, our method is an exemplar-based method. It borrows information from the undamaged region to fill in the damaged region. In other words, the information of the undamaged region in the original image determines the filling information of the damaged region. For example, if we need to fill the damaged region with rich texture information to make the image more natural, but the undamaged region only contains relatively smooth information. In this case, our method can only choose smooth information, which will lead to a reduction in the quality of restoration.

In addition, it should be mentioned that in this article we did not compare our method with the method based on deep learning. We think the basic principles of these two types of methods are different. The exemplar-based method mainly searches for similar exemplar patches from undamaged regions to fill the damaged regions according to certain matching rules, while the method based on deep learning is mainly to use the trained deep network to automatically generate the information in the damaged region. Besides, we have noticed that the methods based on deep learning, especially the methods based on generative adversarial networks can use a large number of real images to train the generative model and the discriminative model, and use the trained generator to automatically generate very realistic and natural images. Some proposed methods have achieved better restoration results on face images. At present, we have begun to study the inpainting method based on the generative adversarial networks, hoping to further improve the restoration effect of large-scale natural scene images.

V. CONCLUSIONS

In view of the problems existing in the image inpainting method for object removal, we propose a method based on adaptive two-round search strategy in this paper. We define the DBP between the exemplar patch and the target patch, and use it to measure the degree of difference between the two patches. Based on the SSD and DBP, we adaptively judge whether there is a mismatch error. If the mismatch occurs, the two-round search strategy is implemented. We define new matching rule based on SSD and DBP and re-research the exemplar according to the rule. Finally the target patch is restored. The experimental results show its effectiveness and feasibility. The next step we will study how to use the generative adversarial networks to further improve the restoration effect of large-scale natural scene images.

REFERENCES

[1] X. Dong, J. Dong, G. Sun, Y. Duan, L. Qi, and H. Yu, "Learning-based texture synthesis and automatic inpainting using support vector machines," *IEEE Trans. Ind. Electron.*, vol. 66, no. 6, pp. 4777–4787, Jun. 2019.

[2] T. Ogawa and M. Haseyama, "Exemplar-based image completion via new quality measure based on phaseless texture features," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 1827–1831.

[3] Q. Guo, S. Gao, X. Zhang, Y. Yin, and C. Zhang, "Patch-based image inpainting via two-stage low rank approximation," *IEEE Trans. Vis. Comput. Graphics*, vol. 24, no. 6, pp. 2023–2036, Jun. 2018.

[4] M. S. Ishi, L. Singh, and M. Agrawal, "A review on image inpainting to restore image," *IOSR J. Comput. Eng.*, vol. 13, no. 6, pp. 8–13, 2013.

[5] W. Hu, Z. Li, and Z. Liu, "Fast morphological component analysis Tangka image inpainting based on nonlocal mean filter," *J. Comput.-Aided Des. Comput. Graph.*, vol. 26, no. 7, pp. 1067–1074, 2014.

[6] I.-C. Chang, J. C. Yu, and C.-C. Chang, "A forgery detection algorithm for exemplar-based inpainting images using multi-region relation," *Image Vis. Comput.*, vol. 31, no. 1, pp. 57–71, Jan. 2013.

[7] Y. Bahat, Y. Y. Schechner, and M. Elad, "Self-content-based audio inpainting," *Signal Process.*, vol. 111, pp. 61–72, Jun. 2015.

[8] L. Zhang, B. Kang, B. Liu, and F. Zhang, "Image inpainting based on exemplar and sparse representation," *Int. J. Signal Process., Image Process. Pattern Recognit.*, vol. 9, no. 9, pp. 177–188, Sep. 2016.

[9] G. Sridevi and S. Kumar, "Image inpainting and enhancement using fractional order variational model," *Defence Sci. J.*, vol. 67, no. 3, pp. 308–315, 2017.

[10] B. V. R. Kumar and A. Halim, "A linear fourth-order PDE-based gray-scale image inpainting model," *Comput. Appl. Math.*, vol. 38, no. 1, pp. 1–21, Mar. 2019.

[11] L. J. Deng, T. Z. Huang, and X. L. Zhao, "Exemplar-based image inpainting using a modified priority definition," *PLoS ONE*, vol. 10, no. 10, pp. 1–18, 2015.

[12] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. 27th Annu. Conf. Comput. Graph. Interact. Techn.*, 2000, pp. 417–424.

[13] J. Shen and T. F. Chan, "Mathematical models for local nontexture inpaintings," *SIAM J. Appl. Math.*, vol. 62, no. 3, pp. 1019–1043, Jan. 2002.

[14] T. F. Chan and J. Shen, "Nontexture inpainting by curvature-driven diffusions," *J. Vis. Commun. Image Represent.*, vol. 12, no. 4, pp. 436–449, Dec. 2001.

[15] F. Yao, "Damaged region filling by improved criminisi image inpainting algorithm for thangka," *Cluster Comput.*, vol. 22, pp. 13683–13691, Mar. 2018.

[16] P. Buyssens, M. Daisy, D. Tschumperlé, and O. Lézoray, "Exemplar-based inpainting: Technical review and new heuristics for better geometric reconstructions," *IEEE Trans. Image Process.*, vol. 24, no. 6, pp. 1809–1824, Jun. 2015.

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

[18] L. Zhang, B. Kang, B. Liu, and Z. Bao, "A new inpainting method for object removal based on patch local feature and sparse representation," *Int. J. Innov. Comput., Inf. Control*, vol. 12, no. 1, pp. 113–124, 2016.

[19] M. Wang, B. Yan, and K. N. Ngan, "An efficient framework for image/video inpainting," *Signal Process., Image Commun.*, vol. 28, no. 7, pp. 753–762, Aug. 2013.

[20] M. Elad, M. A. T. Figueiredo, and Y. Ma, "On the role of sparse and redundant representations in image processing," *Proc. IEEE*, vol. 98, no. 6, pp. 972–982, Jun. 2010.

[21] J. Mo and Y. Zhou, "The research of image inpainting algorithm using self-adaptive group structure and sparse representation," *Cluster Comput.*, vol. 22, pp. 7593–7601, Mar. 2018.

[22] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[23] M. Elad, J.-L. Starck, P. Querre, and D. L. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA)," *Appl. Comput. Harmon. Anal.*, vol. 19, no. 3, pp. 340–358, Nov. 2005.

[24] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, no. 6, 2014, pp. 2672–2680.

[25] Y. Zeng, J. Fu, H. Chao, and B. Guo, "Learning pyramid-context encoder network for high-quality image inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1486–1494.

[26] M.-C. Sagong, Y.-G. Shin, S.-W. Kim, S. Park, and S.-J. Ko, "PEPSI: Fast image inpainting with parallel decoding network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11360–11368.

[27] Y. Jiang, J. Xu, B. Yang, J. Xu, and J. Zhu, "Image inpainting based on generative adversarial networks," *IEEE Access*, vol. 8, pp. 22884–22892, 2020.

[28] J. Wang, K. Lu, D. Pan, N. He, and B.-K. Bao, "Robust object removal with an exemplar-based image inpainting approach," *Neurocomputing*, vol. 123, pp. 150–155, Jan. 2014.

[29] M. Banday and A. Sharma, "A comparative study of existing exemplar based region filling algorithms," *Int. J. Current Eng. Technol.*, vol. 4, no. 5, pp. 3532–3539, 2014.

[30] B. Shen, W. Hu, Y. Zhang, and Y.-J. Zhang, "Image inpainting via sparse representation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2009, pp. 697–700.

[31] Y.-F. Liu, F.-L. Wang, X.-Y. Xi, and Z.-H. Liu, "Improved algorithm for image inpainting based on texture synthesis," *J. Chin. Comput. Syst.*, vol. 35, no. 12, pp. 2754–2758, 2014.

[32] Y.-F. Liu, F.-L. Wang, and X.-Y. Xi, "Enhanced algorithm for exemplar-based image inpainting," in *Proc. Int. Conf. Comput. Intell. Secur.*, 2013, pp. 209–213.

[33] D. J. Florinabel, S. E. Juliet, and V. Sadasivam, "Combined frequency and spatial domain-based patch propagation for image completion," *Comput. Graph.*, vol. 35, no. 6, pp. 1051–1062, Dec. 2011.

[34] J. H. Choi and C. H. Hahm, "An exemplar-based image inpainting method with search region prior," in *Proc. IEEE 2nd Global Conf. Consum. Electron. (GCCE)*, Oct. 2013, pp. 68–71.

[35] Z. Xu and J. Sun, "Image inpainting by patch propagation using patch sparsity," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1153–1165, May 2010.

[36] S. Zhang, K. Wang, and X. Zhu, "Improved criminisi algorithm constrained by local feature," *Comput. Eng. Appl.*, vol. 50, no. 8, pp. 127–130, 2014.

[37] A. Wong and J. Orchard, "A nonlocal-means approach to exemplar-based inpainting," in *Proc. 15th IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 2600–2603.

[38] A. Nan and X. Xi, "An improved criminisi algorithm based on a new priority function and updating confidence," in *Proc. 7th Int. Conf. Biomed. Eng. Informat.*, Oct. 2014, pp. 885–889.

[39] M. Isogawa, D. Mikami, D. Iwai, H. Kimata, and K. Sato, "Mask optimization for image inpainting," *IEEE Access*, vol. 6, pp. 69728–69741, 2018.

[40] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.

LEI ZHANG received the B.S. degree in computer science and technology from Shanxi Normal University, Linfen, China, in 2003, the M.S. degree in computer application technology from Shanxi University, Taiyuan, China, in 2008, and the Ph.D. degree in computer application technology from Northwest University, Xi'an, China, in 2016. His research interests include image processing, pattern recognition, and machine learning.

MINHUI CHANG received the B.S. and M.S. degrees in basic mathematics from Shaanxi Normal University, Xi'an, China, in 2003 and 2009, respectively. Her research interests include image processing, wavelet analysis, and machine learning.