Research Article

Evaluation of Cultural Value Validity of Digital Media Art Based on Locally Weighted Fitting Algorithm

Yuelin Wu, 1 Donghong Li, 2 and Fan Feng 3

1 School of Shijiazhuang Institute of Technology, Shijiazhuang 050020, China
2 Hebei Institute of Communication, Shijiazhuang 050016, China
3 Yihualu Integrated Technology Co., Ltd., Beijing 100043, China

Correspondence should be addressed to Donghong Li; lidonghong20221@163.com

Received 8 April 2022; Revised 10 June 2022; Accepted 22 June 2022; Published 31 July 2022

Academic Editor: Qiangyi Li

Copyright © 2022 Yuelin Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Digital media art differs from other forms of existence in the intersection of digital technology and art, and this new form of existence contains its unique value. Digital media art education crosses the disciplines of art, computer science, and technology and builds a new perspective that can encompass the academic discourse of literature and science, transcending the opposition between scientific and humanistic cultures and creating a broader knowledge structure and a new view of values and resources. In this paper, we use image quality evaluation based on local Gaussian weighted fusion and image collocation for restoration of precious cultural relics and images to realize their cultural values, in view of the current problems such as damage caused by a large number of precious paintings due to improper preservation and other problems.

1. Introduction

The digital virtual world of art calls for the integration of two cultures. While the software framework provides the technical support for the digital virtual art world, artists with traditional literacy must play an integral role in the virtual world. The digital virtual art world requires the artist’s keen insight to draw out the context, and the virtual design needs to be guided by the musician [1]. When art designers do not collaborate with engineers, the construction of virtual worlds loses its foundation. Artists and engineers face the same challenges, and they must work together to build digital spaces.

The purpose of this paper is to realize the value promotion of digital media culture and art. However, there are few researches in this area, so the literature mainly constructs local weighted fitting. Through the study of this treatment, we can provide reference for ourselves. In the digital age of art practice, the two cultures enter a high level of organic fusion, where rigorous technology is subordinated to artistic creativity. Emerging artists do not introduce digital technology into their creations, but rather expand the artistic creativity of digital technology itself. They have created a new artistic language, dedicated to tapping into virtual reality and graphic technology to create works in near-perfect form. They are both users of cutting-edge digital technology and developers of hardware and software. Just as in the 1960s, when the French group “Exploration of Visual Arts” was engaged in the practice of digital media art, there was no such thing as a lone artist or technologist, but rather a team effort that transcended the dichotomy between two cultures [2]. In this paper, we propose a weighted local full-feathering model for image stitching, which can combine multiple images with similar backgrounds into a single panoramic image, and can supplement some missing parts of the image [3]. Firstly, the local firsts of the image are feathered, then the similarity of the grayscale images of the neighboring pixels is optimized by using the method of one-dimensional linear regression, and finally the local firsts of the image are weighted to highlight the block similarity between the pixels in the domain of the image and reduce the traces of feathering. The comparative simulation results of the models show that.

The Bayesian method using Gaussian model as block prior has achieved excellent image denoising performance in
recent years, but the performance of this method is less stable in solving the inverse problem beyond denoising. A hierarchical Bayesian-based Gaussian mixture model is proposed to model image blocks, introduce prior knowledge to the model parameters, and model the probability distributions of the mean and covariance matrices using the Gaussian-Wishart distribution to make the block estimation process more stable [4]. Based on the coherence of neighboring blocks, the clustering of similar blocks in local windows is accomplished using L2 parametric metric, similar blocks in local windows are modeled using multivariate Gaussian probability distributions of specific means and covariances, and the computation time of similarity metric is accelerated using the numerical optimization method of cumulative squared plot and fast Fourier transform. The aggregated weights of Gaussian distribution similarity based on the equine distance are used to combine the spatial domain Gaussian similarity on the images to better fit the statistical properties of natural images. The effectiveness of the proposed model in image restoration solving is verified experimentally [5].

2. State of the Art

With the popularity of image applications, image quality research has attracted a lot of attention. The classical peak signal-to-noise ratio (PSNR) evaluation algorithm has been widely used. In recent years, some quality methods have been proposed based on image engineering theory, and Kipf et al. proposed the structural similarity (SSIM) evaluation method by using the grayscale, contrast, and structural statistics of images to improve the evaluation index of correlation coefficient (CC) of PSNR algorithm. Sheikh proposed the Information Fidelity Rule (IFC) evaluation algorithm based on the Gaussian modeling of natural images [6]. IFC evaluates the image quality based on the amount of information shared between the distorted image and the reference image and is characterized by a parameterless evaluation algorithm [7]. Currently, some improvement algorithms have been proposed for the above algorithms, such as Zhang Lin based on image visual salient features, Ding Yong based on image local statistical characteristics to improve SSIM, Sheikh based on image visual attributes to improve IFC, and Hu Anzhou based on image singular value weight evaluation feature vector to improve M_SVD [8].

However, these improved algorithms have the following problems: (1) the complexity of the algorithm increases, which reduces the real-time evaluation; (2) the performance of some evaluation indexes decreases, such as measurement error and prediction correlation indexes; (3) the algorithm relies heavily on the parameter settings, and the sensitivity of each parameter causes the stability and effectiveness of the algorithm to decrease [9].

Larson improves the index level of the distortion evaluation algorithm through the preprocessing of image frequency domain threshold filtering, and the SSIM algorithm improves the average weight value of image equivalence based on the preprocessing of local Gaussian weighted fusion [10]. Compared with the preprocessing of frequency domain threshold filtering, local Gaussian weighting has the advantages of simple algorithm and high real-time performance. Inspired by this, this paper studies different types of image quality assessment built upon local Gaussian weighted fusion based on the SSIM algorithm [11]. The influence of parameters such as filter window size and filter standard deviation on the performance of the SSIM algorithm and the overall algorithm framework are shown in Figure 1.

3. Methodology

3.1. The Relevant Concepts of the Emerging Concept Filter

Image filtering, that is to suppress the noise of the target image under the condition of preserving the detailed characteristics of the image as much as possible, is an indispensable operation in image preprocessing. Its processing effect will directly affect the effectiveness and reliability of subsequent image processing and analysis.

Due to the imperfections of imaging system, transmission medium, and recording equipment, digital images are often polluted by a variety of noise in the process of their formation, transmission, and recording. In addition, in some links of image processing, when the input image object is not as expected, noise will also be introduced into the result image. These noises often appear as isolated pixels or pixel blocks that cause strong visual effects in the image. Generally, the noise signal is not related to the object to be studied. It appears in the form of useless information and disturbs the observable information of the image. For digital image signals, the noise table is either large or small extreme values. These extreme values act on the real gray value of image pixels through addition and subtraction, causing bright and dark point interference to the image, greatly reducing the image quality and affecting the subsequent work of image restoration, segmentation, feature extraction, image recognition, and so on.

To construct an effective noise suppression filter, two basic problems must be considered: it can effectively remove the noise in the target and background; at the same time, it can well protect the shape, size, and specific geometric and topological features of image targets.

3.2. The Principle of Local Weighted Fusion of SSIM Algorithm

In the present research, the use of fitting is a relatively common method. A lot of literature works show that the feature extraction based on local weighting has better performance than traditional feature extraction. Compared with this method, neural network has been able to achieve better results, so this paper uses deep neural network.

The learning and training of deep neural networks can be realized on many platforms at present. The initial weights of neural networks are randomly generated, and subsequent optimization can be completed by computer braking.

The principle of local weighted fusion of SSIM algorithm is shown in Figure 1. SSIM calculates the local grayscale weighted similarity characteristic $l(x, y)$, local contrast weighted similarity characteristic $c(x, y)$, and structure weighted similarity characteristic $s(x, y)$ of the image,
respectively, and then assigns the corresponding evaluation weights to these three statistical characteristics and performs mean fusion to obtain the evaluation results of distorted images shown in Figure 2.

The local weighted fusion principle of SSIM algorithm is shown in Figure 2. SSIM calculates the local grayscale weighted similarity characteristic \( l(x, y) \), the local contrast weighted similarity characteristic \( c(x, y) \), and the structure weighted similarity characteristic \( s(x, y) \), then assigns the corresponding evaluation weights to these three statistical characteristics, and performs the mean fusion to obtain the evaluation results of distorted images [12].

The definitions of \( l(x, y) \), \( c(x, y) \), and \( s(x, y) \) are as follows:

\[
l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1},
\]

\[
c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2},
\]

\[
s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}.
\]

\[\text{(1)}\]
where the subscripts $x, y$ represent the reference image and the distorted image; the grayscale $\mu$, the variance $\sigma$, and the interrelationship number $\sigma$ are defined as follows:

$$\mu_x = \sum_{i=1}^{N} w_i \cdot x_i,$$

$$\sigma_x = \left( \sum_{i=1}^{N} w_i \cdot (x_i - \mu_x)^2 \right)^{1/2},$$

$$\sigma_{xy} = \sum_{i=1}^{N} w_i \cdot (x_i - \mu_x) \cdot (y_i - \mu_y),$$

where $w$ represents the weighting coefficient of Gaussian filter and $N$ represents the scale size of the square Gaussian filter window. Equations (2) to (4) illustrate that the SSIM algorithm uses local Gaussian weighting coefficients to measure the grayscale and structural similarity of the image. The similarity SSIM $(x, y)$ is defined as follows:

$$SSIM(x, y) = [l(x, y)]^\alpha \cdot [c(x, y)]^\beta \cdot [s(x, y)]^\gamma,$$

where $\alpha, \beta, \gamma$ and $\gamma$ are the weight coefficients corresponding to $l(x, y), c(x, y), s(x, y)$.

The SSIM algorithm only provides the experimental results of Gaussian filter weighting. While the commonly used filters for images are Average, Disk, Gaussian, Laplacian, Laplace Gaussian (Log), Motion Blur, Edge Enhancement (Prewitt), and Edge Extraction. Each filter has control parameters, and different types of filters usually exhibit different performance in different image processing fields. Therefore, in this paper, based on the LIVE image quality evaluation database, we experimentally study the effects of different types of filter weighting operators and their parameters on the performance of SSIM algorithm based on the international video experts group (Video Quality Experts Group, VQEG) specification [13].

### 3.3. Effect of Different Types of Filters on the Performance of SSIM Algorithm

The experiments tested the Root Mean Square Error (RMSE) of the prediction, the CC, and the Spearman Rank Order Correlation Coefficient (SROCC) of the original data [14]. The bold numbers in the table indicate a good level of the index, and the overall evaluation indicates the overall evaluation of all distortion types. Compared with other filters, the stability of the evaluation of distorted images is higher for Average, Disk, and Gaussian filters. The SROCC index shows that the Average and Gaussian filters have better performance; the RMSE and CC indexes show that the Gaussian filter has better performance than the Average filter [15].

### 3.4. Effect of Gaussian Filter Window Scale on the Performance of SSIM Algorithm

Compared with other filters, Gaussian filter weighting has a better overall performance. The effect of the Gaussian filter window scale of $N$ pixels × $N$ pixels on the SS performance is further investigated experimentally, and the results are shown in Figure 2. It can be seen that the SSIM algorithm shows the best performance in terms of RMSE, CC, and SROCC when the window scale is between 80 pixels × 80 pixels and 120 pixels × 120 pixels [16]. The performance of the extreme points of the Gaussian filter window scale is an important guide for the selection of different algorithms in terms of complexity control and evaluation speed as shown in Figure 3.

### 3.5. Effect of Gaussian Filter Data Standard Deviation on the Performance of SSIM Algorithm

In the experiments in Section 3.3, the standard deviation of Gaussian filter $\sigma = 1.5$. In fact, the shape and visual effect of Gaussian filter will be different when the value of $\sigma$ is different [17]. Figure 4 shows that the evaluation indexes RMSE, CC, and SROCC show the best levels when the filter window $N \geq 24$, which correspond to the same standard deviation value ($\sigma = 2$) on average. When the window scale becomes smaller, the index levels all show a certain decrease, and the decrease rate accelerates with $\sigma < 2$ as shown in Figure 4.

The classical PSNR evaluation algorithm based on image pixel statistics has been widely used because of its simplicity and speed, but it has the problems of low correlation and large prediction error compared with the subjective perception of human eyes [18]. Therefore, based on the experimental findings, this paper improves the PSNR and M_SVD algorithms by using partial Gaussian weighted fusion. The improved algorithm first chunks the image with a size of 11 pixels × 11 pixels, then evaluates each subblock based on local Gaussian weighted SNR and M_SVD, and finally performs mean fusion on the evaluation results of each subblock to obtain the final image quality evaluation results [19].

The local peak SNR evaluation algorithm PSNR ($L$) and the local singular value decomposition evaluation algorithm M_SVD ($L$) are defined as follows:

$$PSNR(L) = 10 \times \log \left( \frac{(2^{10} - 1)^2}{MSE(L)} \right),$$

$$M_SVD(L) = \left| D(L) - D_{mid} \right|,$$

$$MSE(L) = \frac{1}{N \times N} \sum_{i=1}^{N} \sum_{j=1}^{N} \left| y(i, j) - x(i, j) \right|^2,$$

$$D(L) = \frac{1}{N - 1} \sum_{i=1}^{N} \left( S_x(i) - S_y(i) \right)^2,$$

where $S(i), S(i)$ are the singular eigenvalues of the reference image and the distorted image, respectively, defined as follows:

$$x(i, j) = U_x \cdot S_x \cdot V_x^T,$$

$$y(i, j) = U_y \cdot S_y \cdot V_y^T.$$
U, V constitute the left and right eigenvectors of the image singular value decomposition.

Based on the image subblock $L$, the MSE ($L$) of PSNR evaluation algorithm and $D$ ($L$) of M_SVD evaluation algorithm are improved by fusion of local Gaussian weighting, respectively, and the following equations are obtained:

$$MSE_g(L) = \frac{1}{N \times N} \sum_{i=1}^{N} \sum_{j=1}^{N} w(i, j)[y(i, j) - x(i, j)]^2.$$  \hspace{1cm} (10)

The improved evaluation results of PSNR ($L$) and M_SVD ($L$) for image subblock $L$ are as follows:

$$PSNR_g(L) = 10 \times \log \left( \frac{2^n - 1}{MSE_g(L)} \right).$$  \hspace{1cm} (11)

$$M_{SVD}_g(L) = [D_g(L) - D_{g_{\text{avg}}}]^2.$$  \hspace{1cm} (12)

where $T$ is the number of subblocks of the image. Table 1 shows the comparison of the evaluation performance of PSNR, M_SVD, PSNR, M_SVD, and SSIM algorithms. It can be seen that the three index levels of PSNR algorithm have been greatly improved, among which the S index level has been improved by 3.78%, which is basically the same as that of SSIM algorithm. The RMSE and CC index levels have been improved by 2.02% and 2.40%, which are obviously more than that of SSIM algorithm. The three MSEs, CC, and SROCC of the improved M_SVD algorithm improved by 4.99%, 0.67%, and 1.78%, respectively.

Table 2 shows the evaluation results of the algorithms based on different distortion levels of images. The mean opinion score (MOS) of distorted images is based on the subjective test specification of the VQEG. From Table 2, we can see that the PSNR algorithm has good stability and meets or exceeds the SSIM algorithm. The M_SVD algorithm has more obvious performance improvement for the evaluation of images with average and high distortion levels [20].

3.6. Real-Time Testing of SSIM Algorithm Based on Gaussian Window Scale. The SSIM algorithm evaluation time based on different Gaussian filter window scales is tested experimentally, and the results are shown in Table 3.

The same filter can be seen such that with the increase of Gaussian filter window scale, the evaluation speed of SSIM algorithm decreases slightly but still maintains a high evaluation speed compared to other algorithms.

4. Real-Time Testing of the Improved Algorithm Based on Gaussian Weighting

The experiments tested the running speed of the improved algorithm after Gaussian weighting, and the results are shown in Table 3. It can be seen that the evaluation speed of the improved PSNR algorithm is comparable to that of the SSIM algorithm, which still maintains a relatively high time performance; the running time of the improved M_SVD algorithm only increases by 6.9%, which still has a large advantage compared with other evaluation algorithms, as shown in Table 4.

4.1. Local Gaussian Mixture Model with the Hierarchical Bayesian Model. In order to apply spatial constraints to images and exploit the coherence of neighborhood blocks, it is assumed that similar blocks in a neighborhood can be derived from a single multivariate Gaussian probability distribution with a specific mean and covariance, letting
each block be associated with a Gaussian model whose parameters are computed from the similar blocks selected in its local nearest neighbor (search window). The nonlocal Bayesian algorithm uses this idea and is applied to the image denoising problem $R$. In this paper, we focus on how the local block model can be applied to more general inverse problem solving, not only to the denoising problem. The main difficulty of this application is the estimation problem of the model, especially when having a high percentage of missing pixels in the degraded image, which is a serious discomfort problem.

In this paper, we propose a hierarchical Bayesian local Gaussian mixture model (IBLGMM), and the general idea of the proposed model is given in Figure 5.

As shown in Figure 5, the similar blocks are computed in the local window using $L_2$ parametrization to make full use of the coherence of the neighboring blocks. Here, it is assumed that the local similar image blocks satisfy a Gaussian prior distribution with mean $\mu$ and covariance $\Sigma$ matrix. A Bayesian hyperprior is introduced for the mean and covariance parameters, and a Norm-Wishart prior is introduced for the image block mean $\mu$ and covariance $\Sigma$ metric (with parameters $\mu_0$ and $\Sigma_0$), using a joint maximum posterior estimation formula to estimate the image block and parameters $\hat{\mu}$, $\hat{\Sigma}$, $\hat{\Sigma}_i$, and $\hat{x}$, $\hat{\mu}_i$, using alternating minimization updates to make the local estimation of Gaussian statistics more stable. In the aggregation from block to image process, for overlapping regions the similarity of blocks for different local Gaussian distributions is used to calculate Gaussian weights based on Mahalanobis distance, and thus the distribution is extended to a Gaussian mixture model. This allows the accuracy of local model estimation to be used for generic recovery problems, especially for missing pixel values (padding problems).

The observed image $Y$ is decomposed into $I$ mutually overlapping blocks $\{y_i\}_{i=1,2,\ldots,I}$, with block size $\sqrt{n} \times \sqrt{n}$. Each block $y \in R^{n \times n}$ can be considered as a re-alization of the random variable $Y$, and $Y$ can be expressed as
\[ Y_i = H_i X_i + V_i, \]  
\( i = 1, \ldots, ITERS \)  
\( H_i \in R^{m \times n} \) is the degeneracy operator; \( X_i \in R^{n 	imes 1} \) is the original block to be estimated; \( V_i \in R^{n 	imes 1} \) is the additive noise term, usually modeled as a Gaussian distribution \( V \sim N(0, \Sigma V) \). The conditional distribution of \( Y_i \) for a given \( X_i \) case is

\[
p(Y_i|X_i) \sim N(H_i X_i, \Sigma V) \propto \text{exp}\left(\frac{1}{2}(Y_i - H_i X_i)^T \Sigma V^{-1} (Y_i - H_i X_i)\right).
\]

The noise covariance matrix \( \Sigma \) assumes a diagonal array and can represent constant variance, spatially varying variance, or variance depending on the pixel values (approximate Poisson noise).

### 4.2. Image Recovery Algorithm

The complete image recovery algorithm for the local Gaussian mixture model with hierarchical Bayesian is shown in Algorithm 2, with the number of iterations set to ITERS. The algorithm requires two stages: minimizing \( f \) by Algorithm 1 and estimating the hyperparameters \( \mu_0 \) and \( \Sigma_0 \). In order to estimate these parameters, it is necessary to rely on the first stage to estimate the image obtained by aggregation of all blocks.

### 4.3. Computational Complexity

In this chapter, the results of two image restoration problems, image denoising and image filling, are presented. The model proposed in this paper is verified by comparing it with some good methods. (In the case of the padding problem, the interpolation is done for two cases: one for randomly observed pixels and another for the zooming problem which can be viewed as an interpolation problem for uniformly observed pixels). The code for the comparison method is taken from the authors’ public code. The experimental images are partly taken from classical synthetic images, such as Lena and Barbara, and partly selected from test images in the BSDS dataset. Some of the experimental images are shown in Figure 5; the first one is a classical synthetic image and the second one is a BSDS image. PSNR is used as an objective measure of the algorithm, and PSNR is defined as

\[
\text{PSNR} = 10 \log \frac{m \times 255^2}{\sum_{i=1}^{m} \xi_i - x_i}.
\]

where \( x \) is the original degraded image, \( \xi_i \) is the estimated image, and \( m \) is the number of pixels in the image. The higher the quality of the estimated image, the higher the PSNR value. The final PSNR value is obtained by averaging 10 times for each experiment.

### 4.4. Parameter Selection

The main parameters involved in the algorithm include the four parameters \( \gamma \), \( \nu \), the prior mean \( \mu_0 \), and the prior covariance \( \Sigma_0 \) of the Norm-Wishart distribution that needs to be set for the calculation of and the parameters for the calculation of the Gaussian weights.

#### 4.4.1. Determination of \( \gamma \) and \( \nu \)

As seen in equation (16), the calculation of the mean \( \mu \) involves the mean estimate of the similarity block \( \sum_{i=1}^{M} T_i Y_i \) and the a priori mean \( \mu_0 \). The parameter \( \gamma \) is associated with the confidence level of the prior \( \mu_0 \), so its value can be a compromise between the confidence of the prior and the information provided by the similar blocks. As the number of similar blocks \( N_{sp} \) increases, the number of known pixels in the block \( N_{sp} \) increases, the information provided by the similar blocks improves. Thus, \( \gamma \) is calculated as follows:

\[
\gamma = \begin{cases} 
N_{sp} \alpha_L, & \text{if } N_{sp} \text{ and } N_{kp} > \text{TH}, \\
N_{sp} \alpha_s, & \text{otherwise},
\end{cases}
\]

where \( \text{TH} \) is the threshold value. A similar approach can be adopted for \( \nu \) (7) plus \( C \) because \( \nu > n - 1 \) as required by the Normal-Wishart prior. The above simple rule has been shown experimentally to be an effective approach. The above approach is mainly aimed at the filling problem, but it needs further investigation in generalizing to other applications of a more general nature as shown in Figure 5.

#### 4.4.2. Determination of \( \mu_0 \) and \( \Sigma_0 \)

\( \mu_0 \) and \( \Sigma_0 \) can be computed from the set of similar blocks of \( C_o \) by the classical maximum likelihood estimation method \( (\bar{x}_i, \bar{x}_{i+1}, \ldots, \bar{x}_M) \):

\[
\begin{align*}
\mu_0 &= \frac{1}{M} \sum_{j=1}^{M} \bar{x}_j, \\
\Sigma_0 &= \frac{1}{M - 1} \sum_{j=1}^{M} (\bar{x}_j - \mu_0)(\bar{x}_j - \mu_0)^T.
\end{align*}
\]

#### 4.4.3. Setting of \( \xi \)

In order to obtain good recovery performance, a small adjustment of \( \xi \) in equation (17) is necessary. For example, in the denoising experiments, different \( \xi \) values are set for different noise levels: at time, \( \xi = 0.015 \); at time, \( \xi = 0.01 \). In the filling experiments, \( \xi = 0.01 \) is set by the small adjustment of \( \xi \), and the PSNR value achieves a certain improvement.

### 4.5. Image Denoising

In the experiments, the original images are contaminated using additive Gaussian noise with standard deviations of 30, 40, 50, 60, and 70, respectively, and in the denoising experiments, \( H \) is the unit array because there are no unknown pixels to be interpolated. In Table 5, the method of this paper (labeled HBLGGMM) is compared with the denoising algorithms such as BM3D, LSSC (learned simultaneous sparse coding), NLB, and EPFL. The values in the table are the average PSNR values (in dB) of all tested images at each noise standard deviation, and the best results are indicated by bolded font.
At all noise levels, the method in this paper is superior to the method of global GMM without applying local constraints (EPLL), as well as the GMM-based method (NLB), from an average point of view. Similar to other global GMM methods, all observation blocks in the group are subtracted as the mean vector of each cluster and added after completing the estimation of MAP, the main purpose of which is to accelerate the execution of the algorithm. The performance of the former GMM-based methods such as PLE and EPLL for image denoising is generally lower than that of the sparse-based methods. The denoising results of this paper are better than the sparse-based methods in the noise standard deviation range from 30 to 60 in terms of average results. The superiority of the LSSC method in terms of PSNR under high noise is related to the advantage of its internal dictionary training, but it is worth mentioning that the online word time of the LSSC method is too long and the practical usability is poor. The example image for comparison is shown in Table 5, and the noise with a standard deviation of 30 is added to the Barbara plot. It is clear from the enlarged area that the method in this paper is more natural in the restoration of pants texture.

Table 5 shows the comparison of the test image denoising results in the BSDS set under high noise (noise standard deviation of 70). From the comparison graphs, we can see that the BM3D and NLB methods have relatively poor image view in this dataset, and the face restoration is not natural enough. The BM3D method has more ripples in the hat area, the EPLL method is not smooth enough in the hat area, the LSSC method has a lot of artifacts in the face and hat area despite the better PSNR value, and the overall view is low. This method is more ideal in both PSNR and visual perception.

<table>
<thead>
<tr>
<th>σ</th>
<th>BM3D</th>
<th>LSSC</th>
<th>EPLL</th>
<th>NLB</th>
<th>HBLGMM</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>29.90</td>
<td>29.86</td>
<td>29.67</td>
<td>29.81</td>
<td>29.93</td>
</tr>
<tr>
<td>40</td>
<td>28.50</td>
<td>28.59</td>
<td>28.18</td>
<td>28.54</td>
<td>28.62</td>
</tr>
<tr>
<td>50</td>
<td>27.61</td>
<td>27.60</td>
<td>27.18</td>
<td>27.32</td>
<td>27.84</td>
</tr>
<tr>
<td>60</td>
<td>25.72</td>
<td>26.74</td>
<td>26.09</td>
<td>26.47</td>
<td>26.74</td>
</tr>
<tr>
<td>70</td>
<td>24.89</td>
<td>26.06</td>
<td>25.43</td>
<td>25.97</td>
<td>25.99</td>
</tr>
<tr>
<td>80</td>
<td>24.20</td>
<td>25.45</td>
<td>24.83</td>
<td>24.99</td>
<td>25.39</td>
</tr>
</tbody>
</table>

4.6. Image Filling. Two application scenarios are designed in the image filling experiment: one is the random masking operator (for random filling), the other is the downsampling masking operator (for image enlargement) under the uniform grid. In the random fill experiment, random masking operators with pixel missing rates of 20%, 30%, 40%, 50%, 60%, 70%, and 80% are set to be applied to the original image. The performance of the proposed method is compared with MCA (morphological component analysis), FOE (field of exports), KR (kernel regression), BP (beta process), and PLE, and the size of the image blocks of all methods is used with a size of 8 x 8 size for all methods. The parameter ξ in the equation used to calculate the cumulative weight which is set to 0.01. The results of comparing this method with other image filling methods are presented in the table, and the values in the table are the average PSNR values of all tested images with different data loss rates. As can be seen from the table, the method in this paper outperforms the comparison methods at all loss rates.

Figures 6 and 7 give a comparison example of local Barbara image restoration with 80% random pixel loss, focusing on comparing the results of smoothing and filling the texture areas of the image. As can be seen from the figure, the other methods in this paper are more effective in both textured and smoothed regions. A special case of image filling is the image resolution scaling problem, which can be seen as an interpolation problem for uniformly sampled images. However, the image magnification problem is more challenging than the filling of randomly observed pixels. Because of this regular sampling, many of the algorithms proposed so far fail to recover the underlying texture in the image well. Random sampling yields better results than uniform sampling. The algorithm in this paper is more robust to the recovery errors caused by uniform sampling.

Table 6 shows comparison of the visual quality of random filling at 80% loss rate. In the experiment, the real high resolution image (high resolution, HR) is downsampled by 2 times to obtain the ground resolution image (low resolution, LR), and after upsampling the LR, uniform sample masking is applied to obtain the degraded image for image filling recovery. The method in this paper is compared with EPLL, Patch Nonlocal, NEDI (new edge-directed interpolation), NARM (nonlocal autoregressive modeling), Bicubic, and other methods.

From Table 6, we can see that the method in this paper has clearer reconstruction results than other methods and can find the real texture better.

4.7. Analysis. With a priori modeling using Gaussian mixture model, it is not possible to use a fixed set of clusters to accurately represent blocks that rarely appear in an image, such as edges or textures. Methods like PLE are actually semilocal (128 x 128 regions) and do not solve this problem. EPLL, on the other hand, is a Gaussian mixture model with 200 components learned from 2 million natural image blocks, which has more mixture components, and EPLL's strategy is more inefficient than the PLE method in the denoising task. In all the above experiments, HBLGMM is superior to the EPLL and PLE methods in both the denoising and the fill and zoom problems. In terms of visual quality,
the details of the image are better reconstructed and the image is clearer. The method in this paper performs estimation from similar blocks in a local window and reinforces the assumption of self-similarity by considering the local neighborhood. This strategy performs neighborhood restriction for the model and therefore makes the estimation more robust. The similarity of the blocks is measured by the similarity of the Gaussian distribution, and the aggregation weights take into account the similarity of the image blocks to the estimated Gaussian distribution, combined with the similarity of the image spatial domain, so that more similar Gaussian blocks will be clustered together by the $L2$
Data Availability

The labeled dataset used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by the School of Shijiazhuang Institute of Technology Hebei Institute of Communication and Yihualu Integrated Technology Co.

References

Advances in Multimedia


