Structure-guided Generative Adversarial Network for Image Inpainting

Huan Liang
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: lianghuan_xatu@163.com

Li Zhao
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: zhaoli1998@163.com

Lei Cao
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: clei0123@163.com

***Abstract*—Generative Adversarial Network based image inpainting algorithms often make errors when filling arbitrary masked areas because all input pixels are treated as effective pixels during convolutional operations. To resolve this matter, we present a novel solution: an image inpainting algorithm that utilizes gated convolutions within the residual blocks of the network. By incorporating gated convolutions instead of traditional convolutions, our algorithm effectively learns and captures the relationship between the known regions and the masked regions. The algorithm utilizes a two-stage generative adversarial restoration network, where the structure and texture restoration are performed sequentially. Specifically, the structural information of the known region in the damaged image is detected using an edge detection algorithm. Subsequently, the edges of the masked area are combined with the color and texture information of the known region for structure restoration. Finally, the complete structure and the image to be restored are fed into the texture restoration network for texture restoration, yielding the complete image output. During network training, a spectral normalization Markovian discriminator is employed to address the slow weight changes during iteration, thereby increasing convergence speed and model accuracy. Based on the Places2 dataset, our experimental findings indicate that our algorithm surpasses existing two-stage restoration algorithms in terms of improving peak signal-to-noise ratio and structural similarity. Specifically, our proposed algorithm achieves a 4.3% enhancement in peak signal-to-noise ratio and a 3.7% improvement in structural similarity when restoring images with various shapes and sizes of damaged areas. Additionally, it produces noticeable visual enhancements, further validating its effectiveness.**

Keywords-Image Inpainting; Edge Detection; Generative Adversarial Network; Gated Convolution; Deep Learning

I. INTRODUCTION

Image inpainting involves the restoration of pixels within a damaged region of an image, aiming to achieve maximum consistency with the original image [1]. It provides various methods and approaches to tackle challenges such as the loss of semantic details, object occlusions, and image content degradation.

During the evolution of image inpainting techniques, traditional machine learning algorithms and deep neural networks have been successively employed and achieved significant progress. With the advancement of deep learning technology, an increasing number of researchers have dedicated efforts to integrating it into the field of image inpainting [2], achieving notable successes. Pathak designed and applied generative adversarial networks on top of traditional convolutional neural networks, proposing encoder-decoder networks [3] and sending network outputs to a discriminator to detect authenticity, significantly enhancing the rationality of results. Nevertheless, the applicability of this network is limited to scenarios involving fixed and regular-shaped masked regions. when confronted with freely-shaped masks, the restoration outcomes may lack the desired level of naturalness. Liu proposed partial convolution to handle irregular holes for image inpainting, masking out ineffective inputs in convolutions and re-normalizing, convolving only with valid pixels, and achieving good restoration results by combining their proposed mask update mechanism. However, as the number of network layer

DOI: 10.2478/ijanmc-2024-0031

increases, it is difficult to learn the relationship between the mask and the image, resulting in mask boundary residues in the restored image. To address these issues, Nazeri proposed a two-stage generative adversarial network image inpainting method that combines edge information priors to accurately reconstruct high-frequency information in images. This approach comprises two key components: an edge restoration network and a texture restoration network. The former predicts the edges within the masked areas of an image, serving as guidance for the latter network, which then proceeds to fill these regions with appropriate textures.

This paper proposes a structure-guided generative adversarial network-based image inpainting algorithm with gated convolution [4] for irregular masked region restoration tasks. The gated convolution facilitates a dynamic feature selection mechanism for the network, adapting to each channel and spatial position. This capability enables the network to choose feature maps in accordance with the semantic segmentation outcomes of particular channels. At the deep layer of the network, gated convolution can also highlight representations of the masking area for different channels. In addition, to ensure stable training, this algorithm employs spectral normalization Markovian discriminators for network generator outputs, providing better restoration results.

II. Related work

The network structure used in this paper is a two-stage generative adversarial restoration network [5], which combines structural and textural restoration to solve image inpainting tasks. This network divides the restoration process into multiple steps. Firstly, the structural information of the known area in the damaged image is obtained through an edge detection algorithm [6]. Then, the boundary of the occluded region is integrated with the color and texture attributes of the known region, culminating in the attainment of structural recuperation. Finally, the complete structure and the image to be restored are inputted into the textural restoration network for textural restoration, resulting in a complete image. The network leverages prior knowledge of image structures to enhance the rationality of the restoration results.

The generator structure in this network consists of two types of convolution: ordinary convolution and dilated convolution combined with residual blocks, designed to broaden the receptive field of convolution. Despite the fact that dilated convolution possesses the capability to augment the receptive field without necessitating an increase in the parameter count, it is prone to losing detailed information when facing small masked areas, resulting in suboptimal performance of the generative adversarial network. To tackle this problem, the paper utilizes gated convolution in place of dilated convolution. This method allows for automatic learning of the mask, enabling the model to capture the connection between the mask and image channels while dynamically adjusting the convolutional receptive field, ultimately enhancing the coherence of the restoration outcomes.

III. Network model structure

The image inpainting network decomposes the restoration task into completion of high-frequency information (edges) and low-frequency information (textures) in the masked area, completing the restoration process in three steps:

Edge detection, which entails the utilization of a comprehensive nested edge detection algorithm to discern the impaired edges within the image. First, the RGB input image I_{in} with defects is converted to a grayscale image I_{gray} with one channel, and then the HED detection algorithm is used to extract the structural information of the image to obtain the edge structure image $E_{\mathrm{d}e}$ with defects.

Structural restoration, which inputs the detected damaged edge image, mask, and damaged image into the structure restoration network. The network includes a generator G1 and a discriminator D1, which outputs the complete edge when the discriminator detects that the generated edge is true. The gray-scale image I_{gray} containing defects, the edge image $E_{\mathrm{d}e}$, and the binary mask image M (with pixel values of 1 for the masked area and 0 for the effective area) are concatenated

along the channel dimension to obtain E_{input}, as shown in Equation (1). E_{input} serves as the joint input of the structure generator G_{edg}.

$$E_{input} = \left\{ I_{gray}, E_{de}, M \right\} \tag{1}$$

As shown in Equation (2), after adversarial training with the edge discriminator D_{edg}, the edge generator outputs the complete edge information E_{co} of the image.

$$E_{co} = G_{edg}\left(E_{input} \right) \tag{2}$$

Texture restoration, which inputs the complete edge and the damaged image to the texture restoration network. The network includes generators G2 and discriminators D2, which output the repaired complete image when the discriminator detects that the filled texture generated by the generator is true. As shown in Equation (3), $\tilde{E}_{co}$ represents the complete structural image inputted into the texture detail generation network. The structural information of the damaged area in $\tilde{E}_{co}$ is the structural generation result of the first stage, and the effective area retains the structural information of the original image. The input of the texture detail generator G_{im} is composed of the damaged image and the edge structure image, denoted as I_{input}.

$$\tilde{E}_{co} = E_{co} \odot M + \left(1 - M\right) \odot E_{de} \tag{3}$$

$$I_{input} = \left\{ \tilde{E}_{co}, I_{in} \right\} \tag{4}$$

The network of the algorithm, as shown in Figure 1, it includes two parts with the same structure: a structure restoration network and a texture restoration network. Each part is a generative adversarial network consisting of a generator with 14 convolutional layers, a discriminator with 6 convolutional layers.

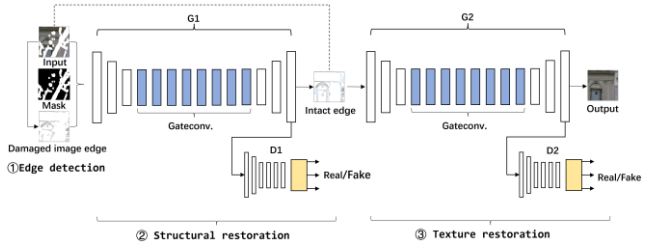

Figure 1. Overall structure of the image inpainting network

A. Generator Network Architecture

The role of the generator is to generate fictitious samples similar to real samples based on real samples, and by continuously improving the reality of generated samples, the discriminator network cannot tell whether an input is a real sample or a fictitious sample. The generators G1 and G2 in the edge restoration network and texture restoration network have the same structure and use gated convolution as the core component of the generator. Specifically, the generator adopts the following structure: the first layer is a normalization layer with 64 convolution kernels of size 7×7 to avoid gradient explosion or disappearance during backpropagation; the second and third layers are downsampling layers that use 128 and 256 convolution kernels of size 4×4 respectively to continuously reduce the image resolution and increase the output receptive field; the fourth to eleventh layers consist of 8 residual blocks, all using 3×3 gated convolution kernels that do not change the image size, and use masked feature filling with gated convolution to reduce gradient disappearance caused by background feature; the twelfth and thirteenth layers are upsampling layers with a size of 4×4, gradually restoring the image to its original resolution; The fourteenth layer consists of an activation function applied after a 7×7 convolutional kernel, designed to mitigate the impact of nonlinearity. Instance normalization is used between each convolutional layer to make each generated sample independent of each other [7].

B. SN-PatchGAN

In order to ascertain the veracity of input data, the discriminator is frequently employed to discriminate between actual samples and synthetic samples produced by the generator. Both D1 and D2 use Spectral Normalization PatchGAN as the

discriminator to determine the authenticity of the generator's restoration results. The training process consists of two steps. First, train the discriminator with a fixed generator. When the input is real data, the confidence is set to 1; otherwise, it is set to 0. While keeping the generator parameters unchanged, maximize the generator loss function value to enable the discriminator to have the ability to distinguish between real and fictitious data. Second, train the generator with a fixed discriminator. While keeping the discriminator parameters constant, minimize the generator loss function value so that the generator can generate images that the discriminator cannot distinguish which one is real. Through the repeated iteration of this minimax game process, the model ultimately achieves a state of equilibrium, thereby stabilizing the training.

The structure of the Spectral Normalization Markovian Discriminator is as follows: 6 convolution layers with a kernel size of 5 and a stride of 2, with 64, 128, 256, 256, 256, and 256 convolution kernels, respectively. By stacking each layer to obtain statistical information of the Markovian block features, it captures different features of the input image in different positions and semantic channels, and directly applies the generative adversarial network loss to each feature element in the feature map.

C. Gated Convolution

The middle layers of the generator network are used to generate features of damaged regions, so continuous residual blocks are needed to maintain gradients during propagation in order to prevent gradient disappearance or explosion. However, conventional residual blocks typically use dilated convolutions, which sacrifice many details associated with known and unknown regions despite obtaining a larger receptive field.

Gated convolution offers a trainable mechanism for dynamically selecting features at each spatial position and channel across all layers, thereby enabling the generalization of partial convolution, thus avoiding the problem of low edge information utilization and lack of relative position information in deep layers caused by partial convolution. Even after multiple rounds of feature extraction and mask updating, the network can still assign different soft mask values to each spatial location based on edge sketch information and whether the current pixel is located in the masked area of the feature image. The structure of gated convolution is shown in Figure 2.

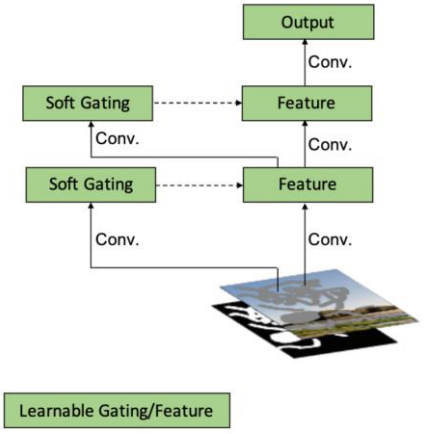

Figure 2. Schematic diagram of gated convolution structure

The gated convolution $O_{y,x}$ consists of the gating selection unit $G_{y,x}$ and the feature extraction unit $F_{y,x}$, as shown in Equations (5)-(7), where I_{f_m} represents the downsampled feature map input in the network.

$$G_{y,x} = \sum\sum W_g \cdot I_{f_m} \tag{5}$$

$$F_{y,x} = \sum\sum W_m \cdot I_{f_m} \tag{6}$$

$$O_{y,x} = \Phi\left(F_{y,x}\right) \odot \sigma\left(G_{y,x}\right) \tag{7}$$

Specifically, the network first calculates the gate value g of the input feature map according to the formula $g = \sigma\left(G_{y,x}\right)$. σ is the sigmoid activation function, which outputs gate values between 0 and 1. W_g is a learnable parameter that serves as a convolution filter used to compute gate values, while W_m is a multi-dilated convolution kernel used for feature extraction from the input image. Φ is the LeakyReLU activation function. The gated convolution structure finally outputs the product of the feature map $F_{y,x}$ and the gate value. Gated convolution enhances the generator's ability to utilize valid elements and edge pixels in the input image, thereby improving its reasoning and

synthesis capabilities for missing regions in images.

D. Loss Function

Structural repair network loss function, To ensure stable and effective training, the loss function of the generative adversarial network in the structure repair network uses the hinge loss to determine the truth or falsehood of the input, including the generator loss L_G and the spectral normalized SN-PatchGAN discriminator loss $L_{D^{sn}}$:

$$L_G = -E_{z\sim Pz}\left(\boldsymbol{z}\right)\left[D^{\text{sn}}\left(G\left(z\right)\right)\right] \tag{8}$$

$$\begin{aligned} L_{D^{sn}} &= E_{\boldsymbol{x}\sim \boldsymbol{P}_{\text{data}}(x)}\left[ReLU\left(1-D^{sn}\left(\boldsymbol{x}\right)\right)\right] \\ &+E_{\boldsymbol{z}\sim \boldsymbol{P}_z(z)}\left[ReLU\left(1+D^{sn}\left(G\left(\boldsymbol{z}\right)\right)\right)\right] \end{aligned} \tag{9}$$

Here, $G(\boldsymbol{z})$ is the output result of the generator G1 repairing incomplete image $\boldsymbol{z}$, and D^{sn} represents the spectral normalized Markov discriminator.

Given that the relevant edge patch information in the image has already been captured in D^{sn}, the use of perceptual loss becomes unnecessary. Instead, a stringent L1 loss function with a substantial penalty is sufficient. Consequently, the final loss function for the structure repair network is composed solely of two components: the pixel-level L1 reconstruction loss L_{rec} and the loss from the spectral normalized Markov discriminator $L_{D^{sn}}$, which are set with a default hyperparameter ratio of 1:1, as shown below:

$$L = L_{rec} + L_{D^{sn}} \tag{10}$$

$$L_{rec}\left(x\right) = M \odot \left(x - \text{F}\left(\left(1-\text{M}\right)\odot x\right)\right) \tag{11}$$

Here, $F(\cdot)$ represents the sampling process of the encoder.

Texture repair network loss function, In the texture restoration stage, a large amount of texture information is filled, causing significant differences in the activation maps of each convolutional layer. To capture the difference in covariance between these activation maps, a style loss is introduced. Given a feature map of size $C_i \times H_i \times W_i$, the expression for the style loss function is:

$$L_{style} = E_i\left[G_i^{\varphi}\left(C_{out}\right) - G_i^{\varphi}\left(I_{in}\right)\right] \tag{12}$$

Here, G_i^{φ} is the $C_i \times C_i$ Gram matrix constructed from the i layer activation map φ_i. The ultimate loss function for the texture restoration network incorporates both the style loss and the SN-PatchGAN loss, configured with a default hyperparameter ratio of 1:1, as detailed below:

$$L = L_{style} + L_{D^{sn}} \tag{13}$$

The expression for $L_{D^{\text{sn}}}$ is the same as Equation (9).

IV. Experiments

A. Experimental Environment

In the experiments, the batch size was set to 8, and both the discriminator and generator learning rates were $1e^{-4}$, with the Adam optimizer (parameters: β_1=0, β_2=0.9) used for network updates. The experimental environment was based on an Ubuntu system with the PyTorch 1.8.3 deep learning framework, and the hardware configuration included a CPU with 128GB of memory and 4 NVIDIA TITAN V GPUs, each with 12GB of VRAM. The proposed improvements were thoroughly tested under the same configuration.

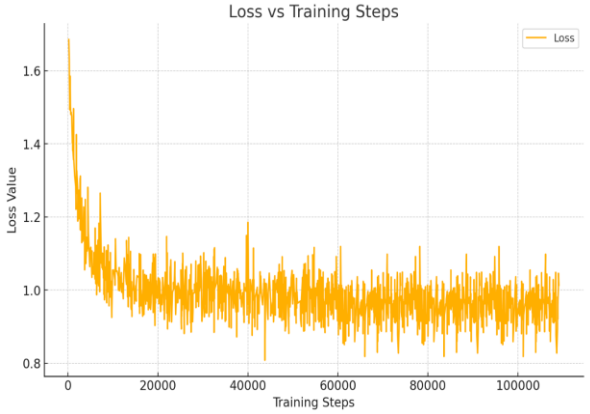

Figure 3. Curves of Loss Functions during Model Training

Figure 3 shows the convergence curves of the loss functions during the training process. As the number of iterations increases, the loss functions of both the generator and discriminator gradually stabilize and eventually converge, completing the training. Throughout the training process, the loss functions of the generator and discriminator are updated alternately, gradually improving the quality of the generated images and enhancing the discriminator's ability to distinguish them. Proper selection of the combination and weights of the loss functions is crucial for training a high-quality GAN model.

B. DataSet

The experimental datasets utilized in this study include the Places2 and CelebA datasets. The Places2 dataset [10] contains approximately 10 million images, and is widely used for image processing tasks related to scenes and environments. The experiments were conducted using the official default training and testing sets. A partial sample of the Places2 dataset is shown in Figure 4. The CelebA dataset, which was publicly released in 2015 by the Chinese University of Hong Kong, is an extensive collection of face attribute data on a large scale. This dataset comprises roughly 202,599 facial images, each accompanied by 40 attribute annotations. A partial sample of the Places2 dataset is shown in Figure 5.

The mask dataset used in this study was contributed by the dataset proposed in [2], which contains 12,000 masked images with mask region ratios ranging from 1% to 90%. During training, the masks were randomly rotated by 0 °, 90 °, 180 °, and 270 °, and horizontally and vertically flipped for data augmentation. To verify and optimize the feature extraction and gating selection capabilities of the gating convolutional layer for different masks, each original image was trained by arbitrarily and repeatedly superimposing random masked areas before being input into the network. A partial sample of the mask dataset is shown in Figure 6.

Figure 4. A partial sample of the Places2 dataset

Figure 5. A partial sample of the CelebA dataset

Figure 6. A partial sample of the Irregular mask dataset

C. Evaluation index

In order to assess the quality of the restoration results, we employed the peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) metrics, as specified in reference [8]. These metrics were employed to calculate the average PSNR and SSIM values for the restored images, where higher scores indicate superior restoration quality. PSNR (peak signal-to-noise ratio) is originally defined as the ratio between the maximum potential signal power and the noise power that impacts its precision. In image processing, PSNR is frequently employed to assess image quality in inpainting tasks. A higher PSNR value signifies less distortion in the compressed image. The corresponding calculation formula is presented below:

$$PSNR = 20 \times lg\left(\frac{\mathrm{MAX_I}}{\sqrt{\mathrm{MSE}}}\right) \tag{14}$$

In this context, $\mathrm{MAX_I}$ denotes the maximum pixel value in the image, while MSE represents the mean squared error between the generated image and the original (noisy) image.

The structural similarity index (SSIM) measures the structural resemblance between an uncompressed, undistorted image and a target image. It assesses similarity across three aspects: luminance, contrast, and structure [9]. Luminance is calculated through the mean value, contrast through the standard deviation, and structural similarity through covariance. A higher SSIM score signifies greater similarity and less distortion, with a maximum value of 1. The formula for its calculation is shown below:

$$SSIM = \frac{(2\mu_X\mu_Y + C_1)(2\sigma_{XY} + C_2)}{(\mu_X^2 + \mu_Y^2 + C_1)(\sigma_X^2 + \sigma_Y^2 + C_2)} \quad (15)$$

Here, μ_X represents the mean pixel value of X, while μ_Y represents the mean pixel value of Y, and the mean value is an estimate of the luminance of the images. σ_X^2 andσ_Y^2represent the variances of$X$and$Y$, respectively, and the standard deviation is an estimate of the contrast of the images. σ_{XY} represents the covariance between X and Y, and it is used as a measure of the structural similarity between the images, with a range from 0 to 1. C_1 and C_2 are constants introduced to ensure stability.

D. Comparative Analysis of Results

In order to verify the effectiveness of the algorithm, the test sets of Places2 and CelebA datasets were used to compare the algorithm with CE, Pconv and EdgeConnect algorithms in terms of subjective results and objective evaluation indicators under different mask region proportions.

Figure 7. The repair effect of each algorithm is displayed

Figure 7 shows the repair results of our method and the comparison methods in each data set. In the first column of the figure, the input image with random mask is added. In the second column to the fifth column, the repair results of CE, Pconv, EC and the algorithm in this paper are respectively applied. The sixth column is the original image.

TABLE I. PSNR/SSIM FOR DIFFERENT IMAGE INPAINTING METHODS AND DIFFERENT MASK AREA RATIOS ON THE PLACES2 DATASET

Mask Ratio	PSNR/SSIM			
	CE	*Pconv*	*EC*	*Ours*
1%-10%	29.26/0.937	30.87/0.929	32.58/0.947	33.89/0.961
10%-20%	21.34/0.746	24.62/0.887	27.15/0.916	28.43/0.935
20%-30%	19.58/0.658	21.43/0.824	24.33/0.859	25.58/0.878
30%-40%	17.82/0.549	19.32/0.751	23.17/0.782	23.81/0.814
40%-50%	15.77/0.475	17.48/0.682	21.64/0.747	22.04/0.763
50%-60%	14.25/0.416	16.44/0.613	19.46/0.651	20.53/0.686

According to the table 1, when the mask area ratio is between 1% and 30%, the peak signal-to-noise ratio of our algorithm ha0073 a significant improvement compared to other algorithms, with an average improvement of about 4.3% compared to the EdgeConnect network. This is because the network uses gate convolution technology to obtain the relationship between the background and the mask, thereby enhancing the consistency and rationality between the known region and the filling region. It also confirms that the two-stage network model has excellent restoration performance. As the mask area ratio gradually increases, the PSNR of all algorithms shows a significant decrease. Nonetheless, the superior performance indicates that the Spectral Normalization Markov Discriminator significantly enhances the network's robustness. When the mask area ratio is between 30% and 60%, the structural rationality of the CE method's restoration effect is poor, and the curve of structural similarity decreases faster. This is because the encoder-decoder network [10] of this method is only suitable for repairing tasks where the mask area is square. Nevertheless, the structural similarity of our proposed algorithm is slightly higher than that of the EdgeConnect method, because the hinge loss function adds a reconstruction loss in the edge recovery process, which constrains the network to generate more complete structural information. This prior information can achieve higher structural similarity results after entering the texture restoration network.

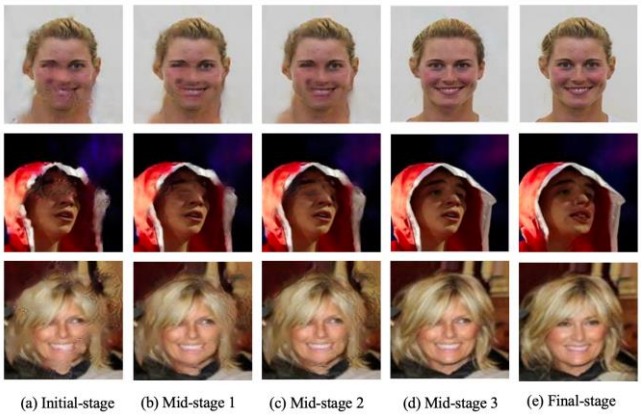

Figure 8. Comparison of Inpainting Results at Different Iterations during Training

Figure 8 shows the comparison of intermediate results generated at different iterations during the deep learning-based image inpainting task. The proposed model demonstrates significantly superior performance compared to other models throughout the training process. In the early stages of training, the generated images exhibit low quality and noticeable blurriness. As the number of iterations increases, the inpainting performance gradually improves, though certain deficiencies remain. During the middle stages, while the overall quality of the restored images improves, localized texture blurring is still apparent. In the later stages of training, despite enhanced overall image quality, texture artifacts and unclear boundary restorations persist. After further iterations, the model achieves a notable improvement in image restoration quality.

Ultimately, the proposed model progressively refines texture details throughout the training process, resulting in final images with sharper visual quality and higher restoration fidelity.

V. Conclusions

To sum up, the present study introduces a novel image restoration algorithm utilizing a gate convolution generative adversarial network. This approach effectively captures the intricate connections between the known and masked regions, enabling the acquisition of meaningful correlations between the image and the corresponding mask. This algorithm effectively improves the quality of image inpainting by solving problems such as unnatural holes and inconsistent filling regions, especially when the mask area ratio is less than 30%. Additionally, using Spectral Normalization Markov Discriminator and hinge loss function can enhance the reconstruction details and stabilize the network training process, thereby improving the speed and accuracy of the algorithm. Future research will focus on texture restoration and try to conduct experiments in content generation of generative adversarial networks to further improve the inpainting effect of the network when repairing images with more than 30% defects.

Reference

[1] Quan W, Zhang R, Zhang Y, et al. Image inpainting with local and global refinement [J]. IEEE Transactions on Image Processing, 2022, 31: 2405-2420.

[2] Wang N, Zhang Y, Zhang L. Dynamic selection network for image inpainting [J]. IEEE Transactions on Image Processing, 2021, 30: 1784-1798.

[3] Qin Z, Zeng Q, Zong Y, et al. Image inpainting based on deep learning: A review [J]. Displays, 2021, 69: 102028.

[4] Navasardyan S, Ohanyan M. The Family of Onion Convolutions for Image Inpainting [J]. International Journal of Computer Vision, 2022, 130(12): 3070-3099.

[5] Zeng Y, Fu J, Chao H, et al. Aggregated contextual transformations for high-resolution image inpainting [J]. IEEE Transactions on Visualization and Computer Graphics, 2022.

[6] Ren Y, Ren H, Shi C, et al. Multistage semantic-aware image inpainting with stacked generator networks [J]. International Journal of Intelligent Systems, 2022, 37(2): 1599-1617.

[7] Yingnan S, Yao F, Ningjun Z. A generative image inpainting network based on the attention transfer network across layer mechanism [J]. Optik, 2021, 242: 167101.

[8] Zhang Y, Ding F, Kwong S, et al. Feature pyramid network for diffusion-based image inpainting detection [J]. Information Sciences, 2021, 572: 29-42.

[9] Moskalenko A, Erofeev M, Vatolin D. Met4hod for Enhancing High-Resolution Image Inpainting with Two-Stage Approach [J]. Programming and Computer Software, 2021, 47(3): 201-206.

[10] Yang Y, Cheng Z, Yu H, et al. MSE-Net: generative image inpainting with multi-scale encoder [J]. The Visual Computer, 2021: 1-13.