


Article

Semantic Image Inpainting with Multi-Stage Feature Reasoning Generative Adversarial Network

Guangyao Li, Liangfu Li * , Yingdan Pu, Nan Wang and Xi Zhang

School of Computer Science, Shaanxi Normal University, Xi'an 710062, China; lgy202649@snnu.edu.cn (G.L.); nhy@snnu.edu.cn (Y.P.); llmunan@snnu.edu.cn (N.W.); tshuajiang@snnu.edu.cn (X.Z.)

* Correspondence: longford@snnu.edu.cn; Tel.: +86-137-5998-6718

Abstract: Most existing image inpainting methods have achieved remarkable progress in small image defects. However, repairing large missing regions with insufficient context information is still an intractable problem. In this paper, a Multi-stage Feature Reasoning Generative Adversarial Network to gradually restore irregular holes is proposed. Specifically, dynamic partial convolution is used to adaptively adjust the restoration proportion during inpainting progress, which strengthens the correlation between valid and invalid pixels. In the decoding phase, the statistical natures of features in the masked areas differentiate from those of unmasked areas. To this end, a novel decoder is designed which not only dynamically assigns a scaling factor and bias on per feature point basis using point-wise normalization, but also utilizes skip connections to solve the problem of information loss between the codec network layers. Moreover, in order to eliminate gradient vanishing and increase the reasoning times, a hybrid weighted merging method consisting of a hard weight map and a soft weight map is proposed to ensemble the feature maps generated during the whole reconstruction process. Experiments on CelebA, Places2, and Paris StreetView show that the proposed model generates results with a PSNR improvement of 0.3 dB to 1.2 dB compared to other methods.

Keywords: deep learning; progressive image inpainting; hybrid weighted merging; point-wise normalization



Citation: Li, G.; Li, L.; Pu, Y.; Wang, N.; Zhang, X. Semantic Image Inpainting with Multi-Stage Feature Reasoning Generative Adversarial Network. *Sensors* **2022**, *22*, 2854. <https://doi.org/10.3390/s22082854>

Academic Editors: Dan Popescu and Loretta Ichim

Received: 18 March 2022

Accepted: 4 April 2022

Published: 8 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Image inpainting aims to reconstruct the missing regions of damaged images and make the repaired image reasonable in both structure and texture. This technology shows a promising performance in many applications, such as image restoration, concealing errors and retouching photos [1–3]. Traditional inpainting methods [4–7] can usually synthesize relatively reasonable stationary textures. However, without semantic understanding of images, they are impossible to generate visually realistic content when the scenes are complex.

Recently, the methods based on encoder-decoder architecture [8–13] have massively improved the inpainting performance; the U-Net structure especially demonstrated a strong ability to generate detailed images. These methods encoded the input image into a latent high-level feature space, and then decoded it back to low-level pixels to fill the missing area in one shot. They are more suitable for filling images with a small range of holes, since the pixels within the local area have a strong correlation. However, as the damaged regions become larger, the model lacks effective features to infer missing contents, one-shot filling will generate semantically ambiguous results. An alternative solution is progressive inpainting. These methods divide the whole restoration process into several phases, each of which employs the information of previous phases as clues to restore the missing area step by step. For example, PGN [14] progressively inpaints the missing regions from the hole boundary to center, but the reconstruction at the image level suffers from high computational cost and information distortion. In order to reduce the amount of calculation and strengthen the connection between features, RFR-Net [15] supposes to share the attention scores of adjacent recurrences. However, there are still several limitations

which could impact the performance of the existing progressive inpainting solutions, and they are summarized as follows. Firstly, current methods ignore the characteristic that the hole regions will gradually shrink. It means that the correlation between valid and invalid pixels will be weakened since the receptive field is fixed which is used to update the mask at different phases. Secondly, the mean and variance of pixels in the valid regions could be different from those of hole regions. Using batch normalization would result in covariant shift. Finally, progressive image inpainting gradually repairs the image by multiple recursions, so early generated signals will corrupt after long term propagation. Using adaptive average merging would degrade the quality of the generated features.

To address these issues, a novel image inpainting framework named Multi-stage Feature Reasoning Generative Adversarial Network (MFR-GAN) is proposed in this paper, which has the ability to generate more realistic and visually pleasing results, as shown in Figure 1. In view of the missing regions gradually narrowing down during the progressive inpainting process, we use the dynamic partial convolution which can regulate the restoration scale according to the scope of damaged areas. By this means, the correlation between known and unknown pixels is enhanced. In the process of progressive image inpainting, the input features will be decoded and encoded multiple times. Since the mean and variance of valid and invalid pixels are different, we designed a new decoding structure which not only leverages point wise normalization instead of batch normalization, but also uses skip-connection to minimize the loss of context information during decoding. To avoid covariant shift as mentioned above, point wise normalization is realized by adaptively assigning scale factors and biases to each feature point in the upsampling process.

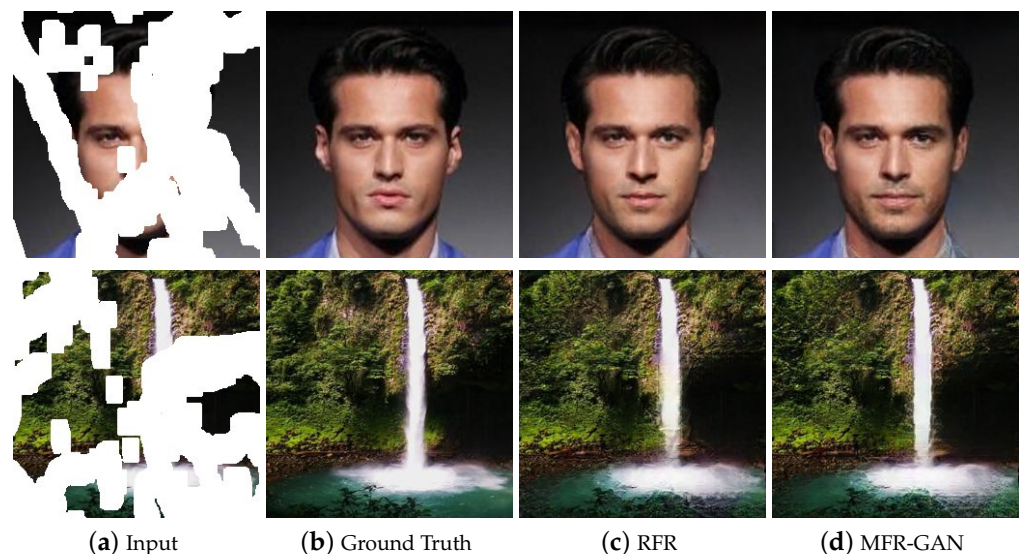


Figure 1. Comparing the proposed method with the state-of-the-art progressive image restoration method [15]. The missing regions are shown in white.

When the reconstruction process is completed, the intermediate information would be ignored if directly using the last recurrence feature map as the final result. Moreover, pixels of the recovered regions are changed during the subsequent inpainting processes. This means that it is difficult to guarantee that the correct clues are always synthesized in intermediate restorations. If wrong information is generated at a certain step, it will be inherited and become worse at the following steps. To tackle these issues, we proposed a hybrid weighted merging method. It is constituted of a hard weight map and a soft weight map. The hard weight map is obtained by analyzing mask characteristics, which could enhance the influence of signals generated in the earlier reconstruction phases. Moreover, our model should be learnable and have the ability to pay attention to certain areas of a reconstructed feature map. To this end, the soft weight map is designed for achieving more realistic content. For the image discriminator, we use the Patch GAN [16] architecture

which enables the model to pay more attention to image details during the training process. The main novelties and contributions of this work can be summarized as follows:

- (1) We design a novel Multi-stage Feature Reasoning Generative Adversarial Network, which can adaptively adjust the inpainting scope in the recursive process through dynamic partial convolution, and leverage point wise Normalization to avoid covariant shift caused by the batch normalization.
- (2) In the image fusion phase, we propose a hybrid weighted merging method that accurately merges the feature maps generated in each recurrence. By this means, we eliminated the problem of the gradient vanishing and the destruction of the content generated in the previous.
- (3) Experiments on the benchmark datasets show that our MFR-GAN has effectively boosted the inpainting performance and generated semantically reasonable content.

The rest of the paper is organised as follows. Section 2 introduces the related work of image inpainting; Section 3 describes methods of the present study; Section 4 is concerned with the main results and the ablation study; Section 5 presents the conclusions.

2. Related Work

Traditional image inpainting methods are mainly composed of two categories: patch-based and diffusion-based. The patch-based inpainting methods [17] filled missing areas by calculating the similarity between patches and transferring similar areas from the background area to the hole area. The diffusion-based inpainting methods [18] attempted to propagate neighboring information to the corrupted areas. Due to capability limitation as well as as lack of semantic understanding of the image, these methods suffer from blurring artifacts when restoring relatively large regions.

Recently, deep learning based methods [19–26] have improved the capability of models to repair complex semantic environments. Context-Encoder [27] firstly employed the deep learning based method, which adopted an encoder-decoder based structure and used GAN [28] for image restoration. Shift-Net [29] introduced a special shift connection layer with the U-Net structure to fill arbitrary masked regions. PEN-Net [30] filled the holes from low resolution to high resolution with a U-shaped pyramid structure to boost the inpainting result. PConv [31] only used valid pixels to infer corrupted regions. GConv [12] further generalized partial convolution to gated convolution that learns to select features for feature maps at each level. Iizuka et al. [32] used a global and local discriminator for adversarial training to obtain coherent filling content. Liu et al. [33] designed region normalization to eliminate the influence of damaged regions on normalization. These methods cannot effectively settle the problems of semantic ambiguity, because they try to reconstruct the entire target without a strong correlation between the hole center pixels and the hole boundary pixels.

SPG-Net [34] factorized image restoration into segmentation prediction and guidance. EdgeConnect [9] utilized the hallucinated edge of the missing area for restoration to ensure structural consistency. Similarly, Xiong et al. [35] used contour and image completion to gradually recover the missing regions. StructureFlow [36] consisted of a texture and structure generator. The structure reconstructor removed high-frequency textures to restore the global structure, and then used appearance flow to synthesize image details. Li et al. [37] designed the visual structure reconstruction layer to restore part edges of a missing area for assisting the completion tasks. Yu et al. [13] devised the contextual attention and leveraged a coarse-to-fine framework to restore damaged images. These methods try to guide image inpainting by adding structural constraints, but they still lack adequate information to reconstruct the central area of the hole. Zhang et al. [14] leveraged U-Net generator with LSTM to concatenate all sub-tasks, and progressively filled the image with the corresponding output sequence. Guo et al. [38] used continuous full-resolution residual blocks to directly fill the missing area of the original size image. Li et al. [15] employed Knowledge Consistent Attention to adaptively combine the attention scores of different recurrent processes to improve the accuracy of image restoration. Although these methods have

achieved considerable progressive, but they are still suffering from limitations described in the introduction.

3. Methods

Figure 2 shows the overall architecture of the Multi-stage feature generation Adversarial network, whose inputs are the damaged image X^{in} and the corresponding binary mask M^{in} , which indicates the missing regions. The proposed model is composed of three components: a feature generator to fill the holes in the feature maps, a feature merging model to accurately fuse the pixels synthesized in every recurrence, and a discriminator used for detail generation. We empirically use two parallel encoders to acquire semantic and image global structure information during the generation process. Firstly, we utilize the dynamic partial convolution to identify the region to be reconstructed in each recurrence. Next, the above operations are performed repeatedly to generate feature maps of different inpainting stages. When the corrupted images are completely filled, the feature maps of each stage are fused by the hybrid weighted merging model to generate the repaired results. Finally, the repaired image is sent to the discriminator for evaluating whether each patch belongs to the real or fake distribution, so as to improve the quality of image inpainting.

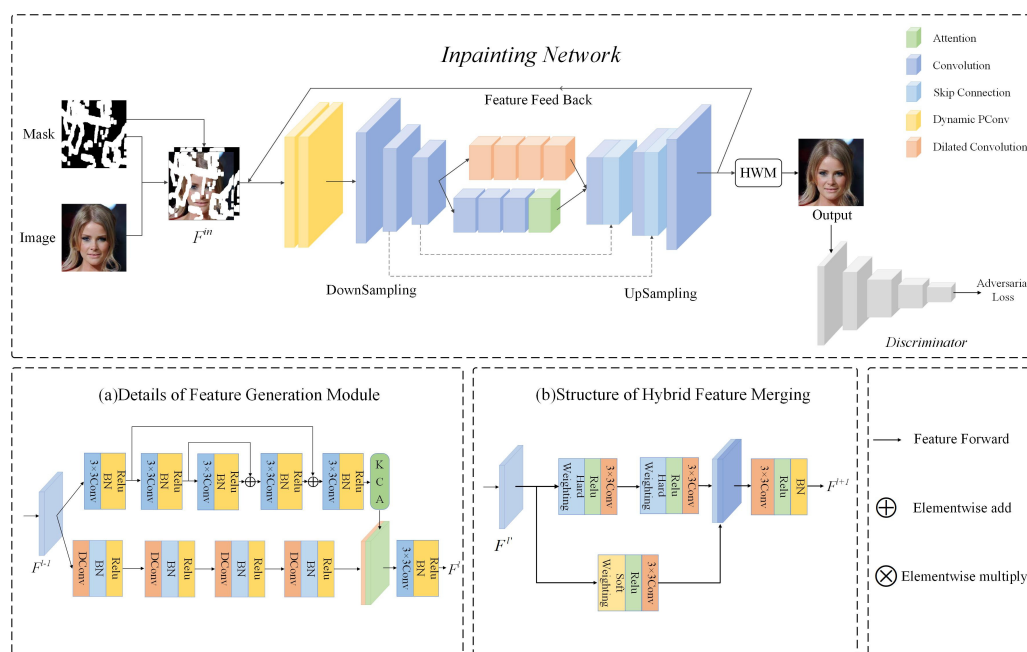


Figure 2. Illustration of our framework. Firstly, we use dynamic partial convolution to update the input mask m^l , then use the inpainting network to generate the pixels for the missing areas. Next, the above operations are performed repeatedly to generate the feature maps of different inpainting stages. Finally, the feature maps of each stage are fused by using the Hybrid weighted merging to generate the repaired results.

3.1. Hybrid Weighted Merging

After a specific number of recurrences, the corrupted image has passed through the feature generation module several times. If we directly use the feature map generated by the last recurrence, it will cause the gradient to vanish and the loss of intermediate generated features. If adopting average merging, the defect area in early reconstructed image will damage subsequently generated information. The early-stage feature information is transmitted farther in the subsequent process to infer the center content; error information may be generated. Therefore, the signals generated in early stages at the same location should be more deterministic and the influence of signals generated in the later recurrences should be decreased. However, adaptively average fusing the valid pixels will affect the early generated feature information. To address this problem, we propose a hybrid weighted

merging method to fuse the feature maps generated during hole restoration processes. It is composed of hard weighting and soft weighting, as shown in Figure 3. For hard weighting, on the premise of fusing non-hole region feature values, the model adaptively generates the weight proportion for current recurrence feature map according to the generation order of valid pixels. Firstly, divide the feature map as shown in the following formula:

$$mask = \begin{cases} mask_i, & \text{if } j = 0 \\ mask_i - mask_{i-1}, & \text{otherwise,} \end{cases} \tag{1}$$

where $mask_i$ is the mask of the i^{th} recurrence update, $mask_{i-1}$ is the mask of the $(i - 1)^{th}$ recurrence update. Then, a weight map W_i is constructed for each recursively generated feature map F^i :

$$W_i^j = \sum_{j=0}^N mask_j \times \frac{1}{1 + e^{(S-i)/(S-j)}}, \tag{2}$$

where i is the recurrence times, j is the number of regions, $mask_j$ is the j^{th} region of $mask_i$. W_i^j represents the j^{th} region of the weight map corresponding to the i^{th} recursive feature map F^i , and S is the number of feature maps generated in this inpainting process. Then, we use the softmax function to generate the proportion of the component $w_{x,y}^i$ for the pixel at position (x, y) . $f_{x,y,z}$ represents the feature value of feature map F^i at location (x, y, z) , and the value of output feature map \bar{F} at location (x, y, z) can be expressed as follows:

$$f_{x,y,z}^h = \sum_{i=0}^S w_{x,y,z}^i \times \delta \left(\frac{f_{x,y,z}^i - \mu_z}{\sqrt{\sigma_z^2 + \epsilon}} \right). \tag{3}$$

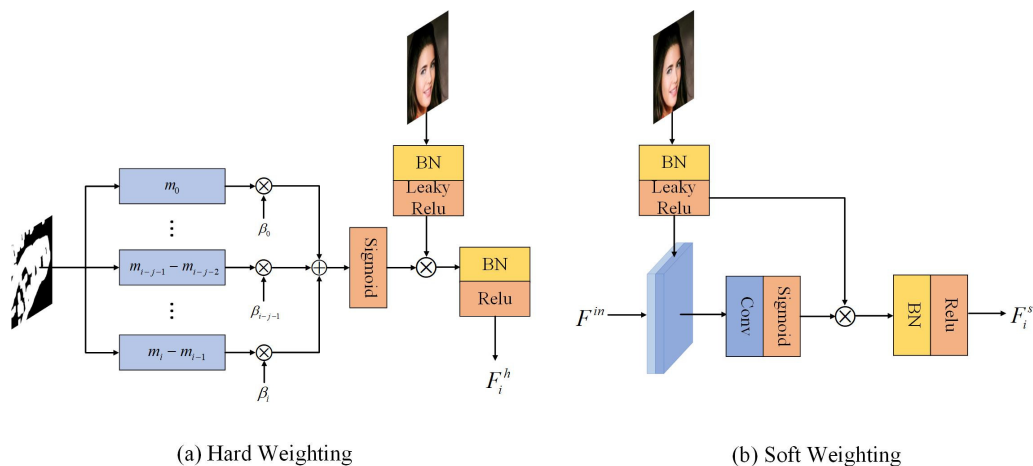


Figure 3. The details of the hybrid weighted merging method.

The hard weight map directly generates the weight by analyzing the mask without the learning process, which would limit the network’s performance. To this end, we propose a soft weight map to assist the hard weighting for achieving better inpainting results. The soft weight map is an adaptive map which is acquired by the input feature and average fused output feature map with a learning process. As shown in Figure 4, we concatenate the F^i and input feature map F^{in} to obtain a soft weight map:

$$W^s = \sigma(\text{Conv}([F^i, F^{in}])) \cdot (1 - M) + M, \tag{4}$$

where σ is the sigmoid activation function and the value of the feature map at location (x, y) after soft weighting can be expressed as follows:

$$f_{x,y,z}^s = \sum_{i=0}^S w_{x,y,z}^s \times \delta \left(\frac{f_{x,y,z}^i - \mu_z}{\sqrt{\sigma_z^2 + \varepsilon}} \right), \quad (5)$$

in which $\mu_z = \frac{1}{H \times W} \sum_{x=1}^H \sum_{y=1}^W f_{x,y,z}^i$, $\sigma_z = \sqrt{\frac{1}{H \times W} \sum_{x=1}^H \sum_{y=1}^W (f_{x,y,z}^i - \mu_z)^2}$, δ represents a leaky relu activation function. By fusing feature maps in the above manner, the gradient vanishing can be effectively avoided, improving the ability of MFR-GAN to restore large damaged areas.

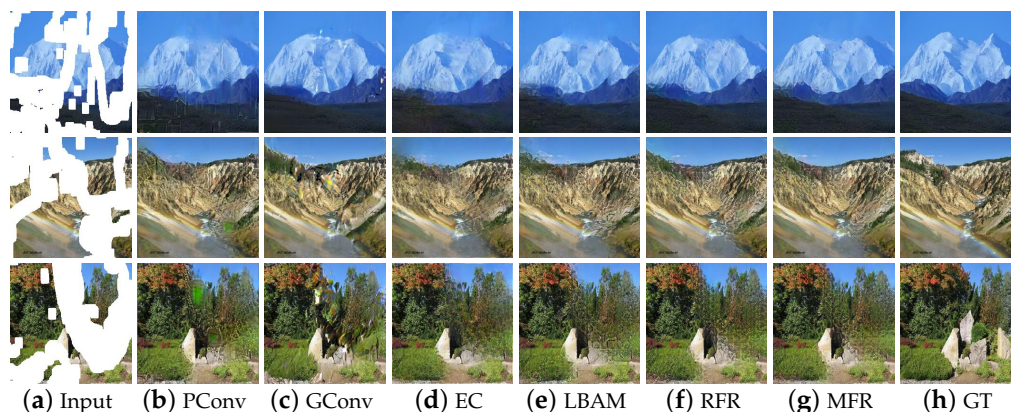


Figure 4. Qualitative comparison on Places2.

3.2. Dynamic Partial Convolution

The significance of the restoration proportion is most noticeable when a model is applied to progressively inpaint the masked image. During the mask updating phase, in order to adaptively identify the area to be repaired in each recurrence, we introduce the dynamic partial convolution. It firstly calculates the holes ratio according to the input mask, and the formula is as follows:

$$D = \frac{\sum_{i=0}^N E_{1,w} \Psi_j^i E_{h,1}}{\sum_{i=0}^N w \times h}, \quad (6)$$

where D is the scale factor, $E_{1,w}$ and $E_{h,1}$ represent the $1 \times W$ row vector and $H \times 1$ column vector with the value of 1. Ψ_j^i represents the mask corresponding to the input image F^{in} , W and H are the width and height of Ψ_j^i , and N represents the value of batch size. Next, the receptive field γ of convolution kernel is obtained by D , its formula is as follows:

$$\gamma = \begin{cases} \lceil D/1e^{-1} \rceil + 2, & \text{if } \beta \% 2 \neq 0 \\ \lceil D/1e^{-1} \rceil + 1, & \text{if } \beta \% 2 = 0 \end{cases} \quad (7)$$

After obtaining the value of γ , the next procedure is updating the area to be repaired. We set stride to 1 and padding = $(\gamma - 1)/2$ ensures that the mask size is consistent with the feature map size, its formula is as follows:

$$f_{x,y,z} = \begin{cases} W^T \left(f_{x,y} \odot m_{x,y} \frac{\text{sum}(1)}{\text{sum}(m_{x,y})} \right) + b, & \text{if } \text{sum}(m_{x,y}) > 0 \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

Here, $f_{x,y,z}$ is the feature value at location (x, y) of layer z ; W^T denotes the weight of the convolution layer filter, $f_{x,y}$ is the input feature patch of the current sliding window,

$m_{x,y}$ is the input mask patch corresponding to $f_{x,y}$, 1 refers to a $H \times W$ matrix with all elements being 1, $\frac{\text{sum}(1)}{\text{sum}(m_{x,y})}$ is the scale factor, the output result is adjusted when the number of convolution effective input values changes. The feature value of the new mask are expressed as:

$$m_{x,y}^* = \begin{cases} 1, & \text{if } \text{sum}(m_{x,y}) > 0 \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

After dynamic partial convolution, the updated mask M^l and the feature map F^l are sent to the feature generation module. The difference between the updated mask and the input mask is defined as the area to be repaired in this iteration, and the updated mask remains unchanged until the next recurrence.

3.3. Feature Generation

A well-designed generator is vital to infer the missing content of the image. In order to fill the hole regions with high-quality features, two parallel encoders are used after down-sampling. The first encoder E_A uses an attention mechanism to synthesize visually realistic textures, and the second encoder E_D uses dilated convolution to collect the spatial features of the feature map. For encoder E_A , we use several convolution layers which are bridged through a skip connection, and apply knowledge Consistent Attention to control the inconsistencies between the adjacent attention feature maps. For encoder E_D , we directly stack four dilated convolution layers. After the input feature maps passing through E_A and E_D , the outputs of the two encoders are concatenated and sent into a single decoder for up-sampling.

3.4. Attention Module

In image inpainting, the attention mechanism can search for possible textures in the background and use them to replace textures in unknown areas. It thus ensures that the filling contents are meaningful in both structure and texture. When the feature map F^i is input into the attention model, we first calculate the cosine similarity between each pair of feature pixels:

$$\widehat{\text{sim}}_{x,y,x',y'}^i = \left\langle \frac{f_{x,y}}{\|f_{x,y}\|}, \frac{f_{x',y'}}{\|f_{x',y'}\|} \right\rangle, \quad (10)$$

where $\widehat{\text{sim}}_{x,y,x',y'}^i$ represents the similarity between features of the background image hole (x, y) and the foreground image hole (x', y') . Then utilizing the similarity of the target pixels in the adjacent areas, we carry out $k \times k$ filtering to smooth the attention score:

$$\text{sim}_{x,y,x',y'}^i = \frac{\sum_{p,q \in \{-k, \dots, k\}} \widehat{\text{sim}}_{x+p,y+q,x',y'}}{k \times k}. \quad (11)$$

After that, the softmax function is used to generate the attention score, which is expressed as score^l . If the features at position (x, y) are valid in the last iteration, their attention scores are adaptively combined with present scores to synthesize the current iteration score:

$$\text{score}_{x,y,x',y'}^i = \text{score}_{x,y,x',y'}^{i-1}. \quad (12)$$

If the pixel value at the (x, y) is invalid in the previous recurrence, the attention score obtained in this recursion is the final score:

$$\text{score}_{x,y,x',y'}^i = \lambda \text{score}_{x,y,x',y'}^{i-1} + (1 - \lambda) \text{score}_{x,y,x',y'}^{i-1}. \quad (13)$$

Next, the attention score is used to reconstruct the feature map \widehat{F} . The feature value of \widehat{F} at (x, y) is $\widehat{f}_{x,y}$, and the calculation process of $\widehat{f}_{x,y}$ is as follows:

$$\hat{f}_{x,y}^i = \sum_{x' \in 1, \dots, W' \in 1, \dots, H} \text{score}_{x,y,x',y'}^i f_{x',y'}^i. \quad (14)$$

Finally, after splicing feature map \hat{F} and input feature map F^{in} , a pixel-by-pixel convolution is performed to generate the reconstructed feature map F' .

3.5. Point Wise Normalization

After the recovery decoder, the generated feature maps are sent for up-sampling. We infer that the statistical characteristics of pixels in the hole regions and the pixels in the no-hole regions are different. Using traditional batch normalization could ignore this characteristic and cause the covariant shift. To tackle this issue, we utilize point wise normalization in the decoding phase to dynamically produce the mask aware scale and bias of batch normalization. The input feature is first normalized in the channel wise manner, and then modulated with learned scale and bias.

$$\bar{f}_{x,y,z}^i = \gamma_{x,y,z}^i(m^i) \frac{f_{x,y,z}^i - \mu_z^i}{\sigma_z^i} + \beta_{x,y,z}^i(m^i), \quad (15)$$

where $f_{x,y,z}$ is the feature before normalization, μ_z and σ_z are the mean and standard deviation of the activation in channel z .

3.6. Loss Function and Model Architecture

The entire training procedure is illustrated in Algorithm 1. We use a Patch-GAN [16] discriminator for image restoration learning. The Patch-GAN discriminator calculates the adversarial loss from the generator. The loss function is consisted of L1 loss, perceptual loss, style loss, adversarial loss and TV loss. L1 loss ensures the accuracy of feature map reconstruction. Given the binary mask with zeros indicating missing pixels, we define the L1 loss as follows:

$$L_{hole} = \frac{1}{N_{I_{gt}}} \|(1 - M) \odot (I_{out} - I_{gt})\|_1 \quad (16)$$

$$L_{valid} = \frac{1}{N_{I_{gt}}} \|M \odot (I_{out} - I_{gt})\|_1, \quad (17)$$

where I_{gt} and I_{out} is the ground-truth image and output value of the network. $N_{I_{gt}}$ is the total number of elements in the image, which equals $C \times H \times W$.

The perceptual loss proposed by Gatys et al. [39] is used to force the filled image and the ground-truth image have similar feature representation. It can be written as follows:

$$L_{perc} = \sum_{n=0}^{N-1} \frac{\|\Psi_n^{I_{out}} - \Psi_n^{I_{gt}}\|_1}{N_{\Psi_n^{I_{gt}}}} + \sum_{n=0}^{N-1} \frac{\|\Psi_n^{I_{com}} - \Psi_n^{I_{gt}}\|_1}{N_{\Psi_n^{I_{gt}}}}, \quad (18)$$

where Ψ_n represents the n^{th} feature layer select in the fixed VGG, I_{com} is composed of the hole range pixels of raw output image and non-hole pixels of the ground truth image. $N_{\Psi_n^{I_{gt}}}$ is the number of elements in $\Psi_n^{I_{gt}}$ and is used as a normalization factor.

A VGG-based [40] Style loss is similar to perceptual loss. The autocorrelation of each feature map is calculated before measuring the L1 distance, the computation of the style loss is as follows:

$$L_{style_{out}} = \sum_{n=0}^{N-1} \frac{1}{C_n C_n} \left\| K_n \left(\left(\Psi_n^{I_{out}} \right)^T \left(\Psi_n^{I_{out}} \right) - \left(\Psi_n^{I_{gt}} \right)^T \left(\Psi_n^{I_{gt}} \right) \right) \right\|_1 \quad (19)$$

$$L_{style_{com}} = \sum_{n=0}^{N-1} \frac{1}{C_n C_n} \left\| K_n \left(\left(\Psi_n^{I_{com}} \right)^T \left(\Psi_n^{I_{com}} \right) - \left(\Psi_n^{I_{gt}} \right)^T \left(\Psi_n^{I_{gt}} \right) \right) \right\|_1. \quad (20)$$

Algorithm 1: Multi-stage Feature Reasoning GAN

Input:
 Img_{in} : Input image
 $Mask_{in}$: random masks

Output:
 Img_{rep} : Repaired image

- 1 $X^0, M^0 \leftarrow DPconv(Img_{in}, Mask_{in})$
- 2 $FeatureGroup \leftarrow \{X^0\}$
- 3 $i \leftarrow 0, IterNum \leftarrow 7$
- 4 **while** i smaller than $Iternum$ **do**
- 5 $x^1, m^1 \leftarrow DPconv(X^0, M^0)$
- 6 $x^2, m^2 \leftarrow DPconv(x^1, m^1)$
- 7 $x^3 \leftarrow DownSampling(x^2)$
- 8 $x^4 \leftarrow x^3$
- 9 $x^5 \leftarrow KCA(x^3), x^6 \leftarrow DilatedConvolution(x^4)$
- 10 $x^7 \leftarrow Cat(x^5, x^6)$
- 11 $x^8 \leftarrow Conv(x^7)$
- 12 $X^i \leftarrow UpSampling(x^8)$
- 13 $FeatureGroup \leftarrow FeatureGroup + \{X^i\}$
- 14 $i \leftarrow (i + 1)$
- 15 **end**
- 16 $X_{merged} \leftarrow HybridWeightedMerging(FeatureGroup)$
- 17 $Img_{rep} \leftarrow DeCoding(X_{merged})$
- 18 $Discriminator \leftarrow Img_{rep}$
- 19 **return** Img_{rep}

In (19) and (20), K_n is the normalization factor for the p^{th} , equal to $1/C_n H_n W_n$. The final loss term total variation loss is expressed as follows:

$$L_{tv} = \sum_{(i,j) \in R, (i,j+1) \in R} \frac{1}{N_{I_{com}}} \left\| I_{com}^{i,j+1} - I_{com}^{i,j} \right\| + \sum_{(i,j) \in R, (i+1,j) \in R} \frac{1}{N_{I_{com}}} \left\| I_{com}^{i+1,j} - I_{com}^{i,j} \right\|, \quad (21)$$

where $N_{I_{com}}$ represents the number of pixels in I_{com} . Total loss L_{total} is the combination of the above loss functions:

$$L_{total} = \lambda_{valid} L_{valid} + \lambda_{hole} L_{hole} + \lambda_{perc} L_{perc} + \lambda_{style} L_{style} + \lambda_{adv} L_{adv} + \lambda_{tv} L_{tv}. \quad (22)$$

In this paper, the weight parameters of each loss function in Equation (22) are set as follows: 1 for λ_{valid} , 6 for λ_{hole} , 0.1 for λ_{tv} , 120 for λ_{style} , 0.1 for λ_{adv} and 0.05 for λ_{perc} .

4. Experiments

This section starts with the introduction of detailed experimental settings, then we compare our model with other methods in terms of both visual quality and quantitative measurements to demonstrate the efficiency of our proposed method. Finally, we conduct an ablation study to examine the design details of MFR-GAN.

4.1. Datasets

We used three well-known public image datasets and a mask dataset [31] to verify our model. Images were resized to 256×256 for training and testing.

- Places2 Dataset [41]: a dataset containing over 365 scenes, which enables the model to learn the distribution from many natural scenes. Including 1.8 million images for training and 12K images for testing.

- CelebA Dataset [42]: a dataset that focuses on face images, containing about 200K images. We used more than 180K images training models, and the other images were used for testing.
- Paris StreetView [43]: a dataset collected from Street View in Paris that is commonly used for inpainting methods. It contains 14,900 images for training and 100 images for testing.

4.2. Training Settings

For the MFR-GAN, we used the Adam optimizer to optimize the generator and discriminator. We trained the model with a batch size of 6 on 2 11G NVIDIA RTX2080Ti GPUs. At the beginning, we trained the model with a learning rate of 1×10^{-4} . Then we set the learning rate to 1×10^{-5} for fine tuning the model, and it was kept unchanged until the model convergence. For CelebA, we first trained 200 K times, then fine tuned 500 K times until convergence. Pytorch was used as the deep learning development framework, and the CUDA version was v10.0.

4.3. Comparison Method

We compared our model with several recent state-of-the-art methods. These models are: PConv [31], GatedConv [12], EdgeConnect [9], LBAM [44] and RFR [15]. For LBAM, EdgeConnect, RFR-Net and GConv, we directly used the officially released pretrained model. Since the source code of partial convolution was not available, we implemented it with the experimental settings in the paper. Compared with PConv, GConv, LBAM and EC, the proposed method strengthens the constraint on the center of the hole by progressively reasoning the content of the hole from the edge of the hole. Compared with RFR-Net, the proposed method solves the problem of covariant shift by using point-wise normalization, and a hybrid fusion method is introduced to make full use of the feature maps generated at each stage, which effectively improves the ability of the model to repair large holes.

4.4. Qualitative Comparisons

For qualitative comparisons, we compared the irregular holes inpainting results of our method with five existing methods on the Places2, CelebA and Paris StreetView datasets. Figure 4 shows that there are varying degrees of blurry boundary and distorted structures when PConv and Edgeconnect repair natural images. Moreover, when PConv repairs the semantic complex natural image, the color distortion is relatively serious, and neither structural connectivity of the generated content nor smooth and reasonable color content can be achieved. GConv well preserves the source image contents, but there are still color inconsistencies in some areas of the result. The result of LBAM suffers from center blur due to the lack of information for restoring deeper pixels in holes. Although RFR-Net can generate meaningful content through leveraging the learned intermediate signals for further restoration, the results still have unreasonable textures. In contrast, the results generated by our model have reasonable semantics and visual authenticity.

Figure 5 shows the inpainting results on CelebA; it is observed that content synthesized by PConv, LBAM and GConv is relatively vague, and the color of some areas is different from the original image. When the holes are relatively large, the image edge generated in the first stage of EdgeConnect includes error information, which leads to the failure of generating a correct structure in the second stage. RFR-Net can generate a plausible structure, but the results still contain unreasonable textures. Through the flexible combination of point wise normalization and skip-connection during the decoding process, our model could make full use of contextual feature information, and generates the content with reasonable semantics and rich details.

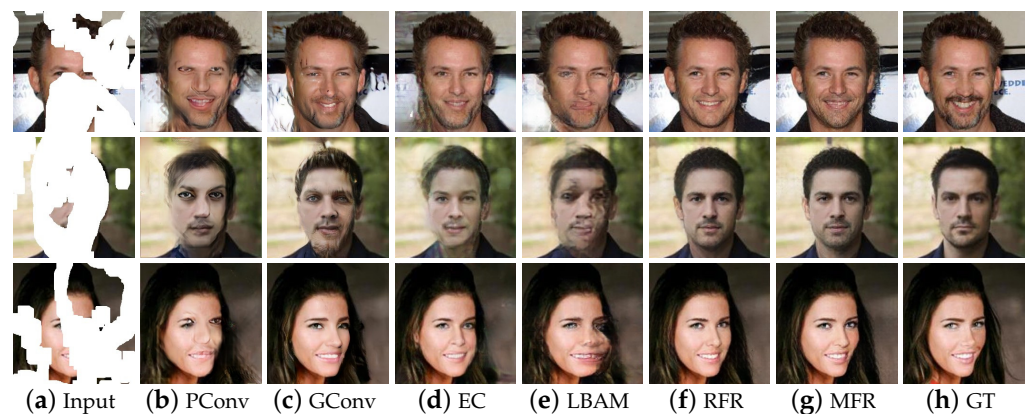


Figure 5. Qualitative comparison of CelebA.

The Paris StreetView Dataset contains images of highly complex structures. As shown in Figure 6, when too much valid information is missing, there are obvious artifacts in the inpainting results of PConv and GConv. LBAM, EC and RFR can generate natural structures, but the results still have unsmooth content. Compared with the above methods, the texture of the inpainted regions is more natural in our results. These results suggest that our method can learn to synthesize better signals by making full use of mask information to gradually fill the missing contents. Besides, in the feature fusion phase, on the basis of eliminating gradient vanishing, the hybrid weighted merging method enables the model to adaptively fuse the feature maps with a learning process, which greatly enhances the reasoning capability of the model.

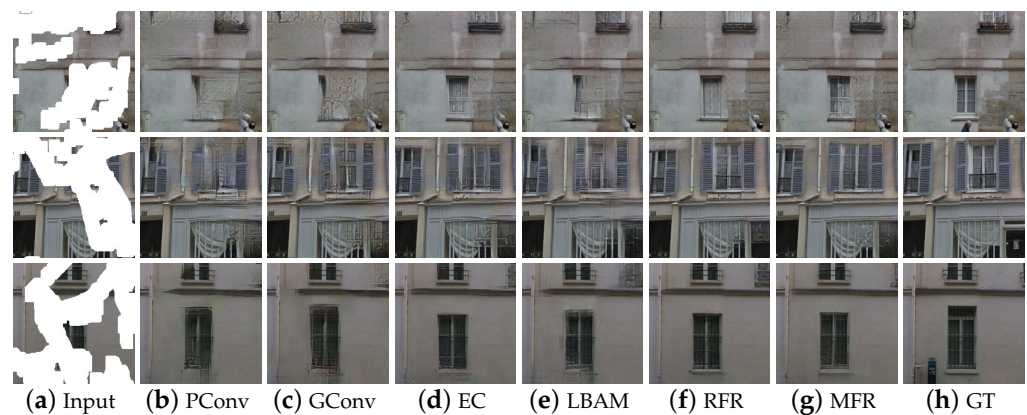


Figure 6. Qualitative comparison on Paris StreetView.

4.5. Quantitative Comparisons

Compared with other datasets, Places2 contains more scenarios, thus it can better verify the authenticity of different methods. For the evaluation metrics, we use peak signal-to-noise ratio (PSNR) to measure the L2 distance. The larger the value of PSNR, the better the image restoration effect is. Usually, when the PSNR value is greater than 28, there is no significant difference in image quality. SSIM is used to measure structural similarity; its value range is between 0 and 1. The higher value means less image distortion. Fréchet Inception Distance (FID) is used to measure the Wasserstein-2 distance between fake and real images. A lower FID score indicates that the two sets of images are more similar, and a score of zero in the best case indicates that the two sets of images are the same. We use the same irregular mask as PConv for testing. The masks are divided into six categories according to the proportion of holes: $(0.01, 0.1]$, $(0.1, 0.2]$, $(0.2, 0.3]$, $(0.3, 0.4]$, $(0.4, 0.5]$, $(0.5, 0.6]$. For each category, we randomly use 50 masks and images as input to test the model. For the performance of repairing irregular masks, as shown in Table 1, when the missing ratio is $(0.2, 0.3]$, the average PSNR of the repair results is 26.23 dB,

the SSIM is 0.922, and the FID is 12.79. Our method has produced excellent results. When the hole ratio is (0.5, 0.6], the PSNR is 19.53 dB, the SSIM is 0.669, and the FID is 34.27. Although the effective information is insufficient, our model can still generate clear content through multiple inferences, as shown in Figure 7 and Table 1. This further validates the effectiveness of our method.

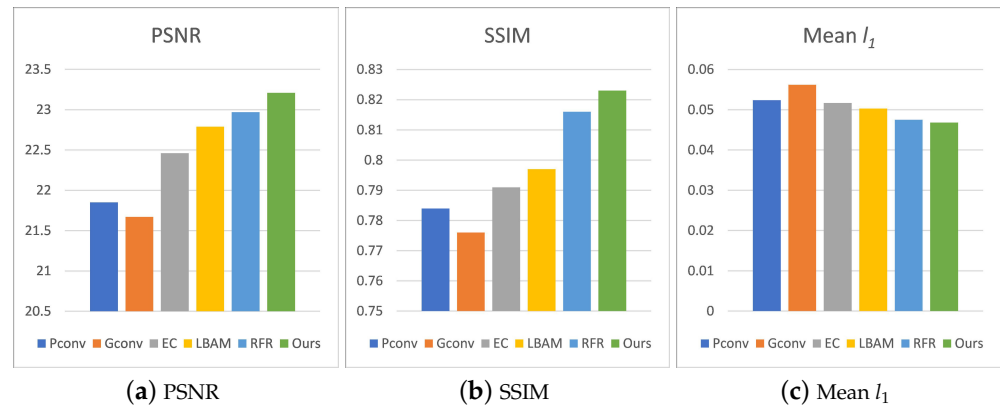


Figure 7. Quantitative comparison of CelebA, the mask ratio is between 0.5 and 0.6.

Table 1. Numerical comparisons on the Place2 dataset. \uparrow indicates higher is better while \downarrow indicates lower is better.

Methods		(0.01, 0.1]	(0.1, 0.2]	(0.2, 0.3]	(0.3, 0.4]	(0.4, 0.5]	(0.5, 0.6]
PSNR \uparrow	PConv	34.15	27.68	25.23	22.34	21.35	18.71
	GC	34.03	27.12	24.95	22.13	20.92	18.35
	LBAM	34.62	27.87	25.42	22.91	21.75	18.81
	EC	34.36	27.46	25.58	22.83	21.58	18.97
	RFR-Net	34.94	28.05	25.99	23.56	21.78	19.36
	Ours	35.31	28.29	26.23	23.83	21.95	19.53
SSIM \uparrow	PConv	0.982	0.943	0.901	0.839	0.787	0.632
	GC	0.981	0.939	0.898	0.835	0.779	0.618
	LBAM	0.984	0.947	0.905	0.846	0.783	0.634
	EC	0.983	0.942	0.908	0.848	0.789	0.636
	RFR-Net	0.985	0.948	0.915	0.871	0.805	0.663
	Ours	0.987	0.954	0.922	0.877	0.811	0.669
FID \downarrow	PConv	4.56	11.21	16.63	23.15	34.27	47.59
	GC	4.24	9.79	15.74	24.92	33.94	45.13
	LBAM	4.12	9.35	14.85	22.41	28.54	42.31
	EC	3.97	8.27	14.59	19.57	26.94	39.15
	RFR-Net	3.28	7.03	13.45	18.34	24.38	35.82
	Ours	2.93	6.81	12.79	16.85	23.26	34.27

4.6. Ablation Study

In this section, we conduct ablation experiments on the Places2 datasets, and illustrate the effectiveness of dynamic partial convolution, hybrid weighted merging, point wise normalization and the effect of recurrence number based on PSNR, SSIM, and FID.

Dynamic partial convolution As demonstrated in Figure 8, we visualize the changes of mask during the whole restoration process. The first row and third row show the mask variation of our model during the inpainting process, and the second row and fourth row are RFR-Net. It is shown that the hole change in our method is smoother than that of RFR-Net. Specifically, our model has learned to dynamically adjust the size of receptive field according to the mask ratios for updating the mask, which is beneficial for strengthening the connection of corrupted area pixels and valid pixels during the progressive completion process.

Hybrid weighted merging The hybrid weighted merging method effectively increases the reasoning times. For large holes, the model can perform multiple inferences for generating more realistic content. Figure 9 shows the visualization of the hard weight map and the soft weight map. The weights of the hard weight map are obtained by the generated order of the signals, as shown in Figure 9a; because the center is repaired in the later recurrence, thus the weight gradually increases from the hole boundary to the center. As shown in Figure 9b, the weight distribution of the soft weight map is uniform, and the weights are adaptively changed by borrowing information from the input features. BN+G AW is a U-Net architecture using average merging, which achieves a PSNR of 21.83 dB. We replace the average merging with hybrid weighted merging, which is named BN+G HW. As shown in Table 2, under the same training strategies and experimental settings, BN+G HW achieves an average PSNR of 21.89 dB, SSIM of 0.808 and FID of 23.82. We can see that the quantitative score is significantly improved after using the hybrid weighted merging method.

Table 2. Quantitative ablation study on the Places2 dataset.

Method	PSNR	SSIM	FID
BN+G AW	21.83	0.806	24.21
BN+G HW	21.89	0.808	23.82
PN+G AW	21.92	0.809	23.54
PN+G HW	21.95	0.811	23.26

As shown in Figure 10, we compared the hybrid weighted merging method with other feature fusion methods on the Paris StreetView dataset, which has more repetitive structures such as gates. It can be observed that the image structure generated by our method has clear textures and consistent contextual structures.

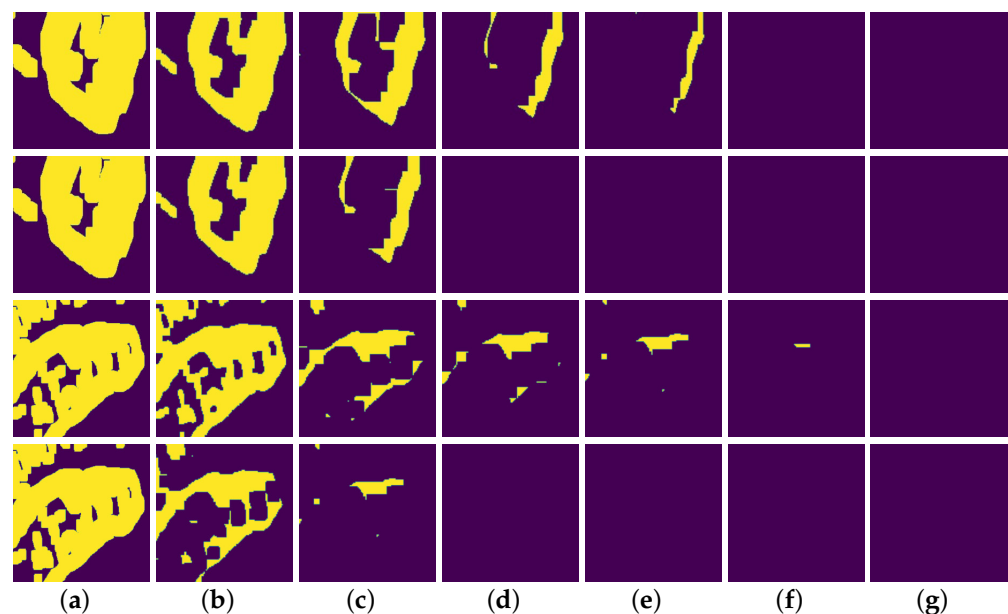


Figure 8. Visualization of the mask updated during the inpainting process. Yellow areas represent invalid pixels, and purple areas denote valid pixels. From the left to the right are: (a) Input mask, (b) first updated mask, (c) second updated mask, (d) third updated mask, (e) fourth updated mask, (f) fifth updated mask, (g) sixth updated mask.

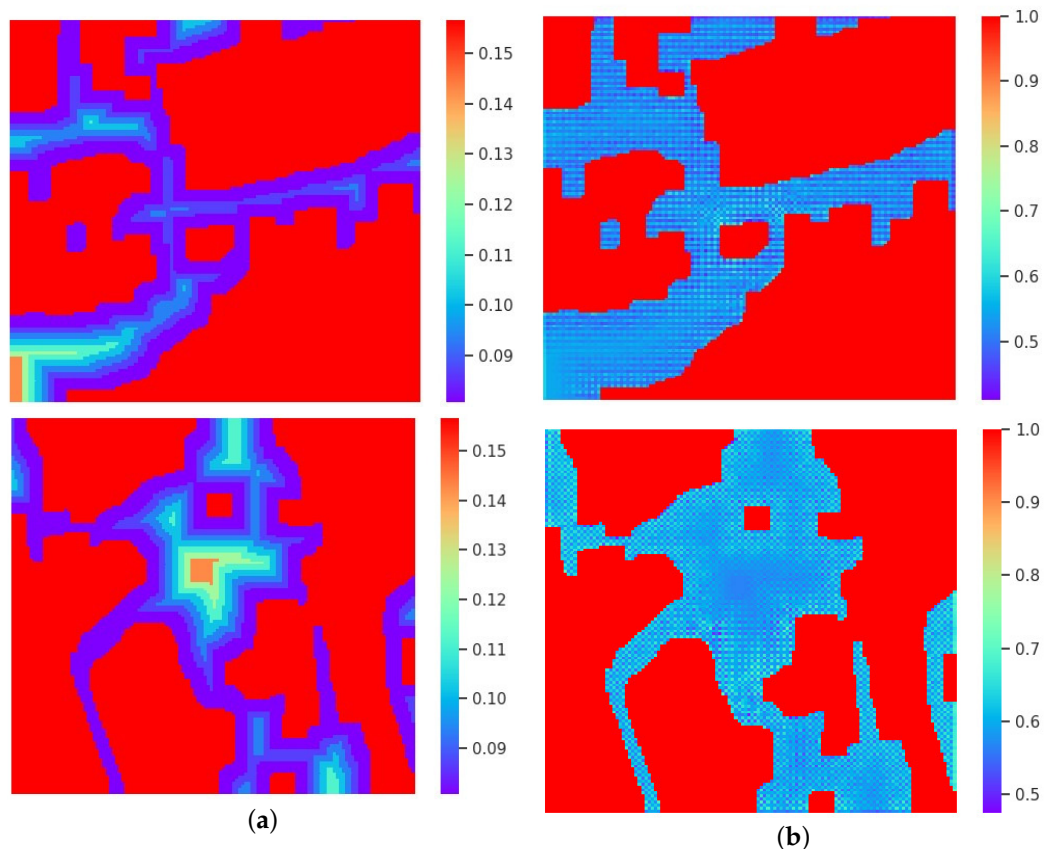


Figure 9. The visualization of the hard weight map and soft weight map. (a) is the hard weight map and (b) is the soft weight map.

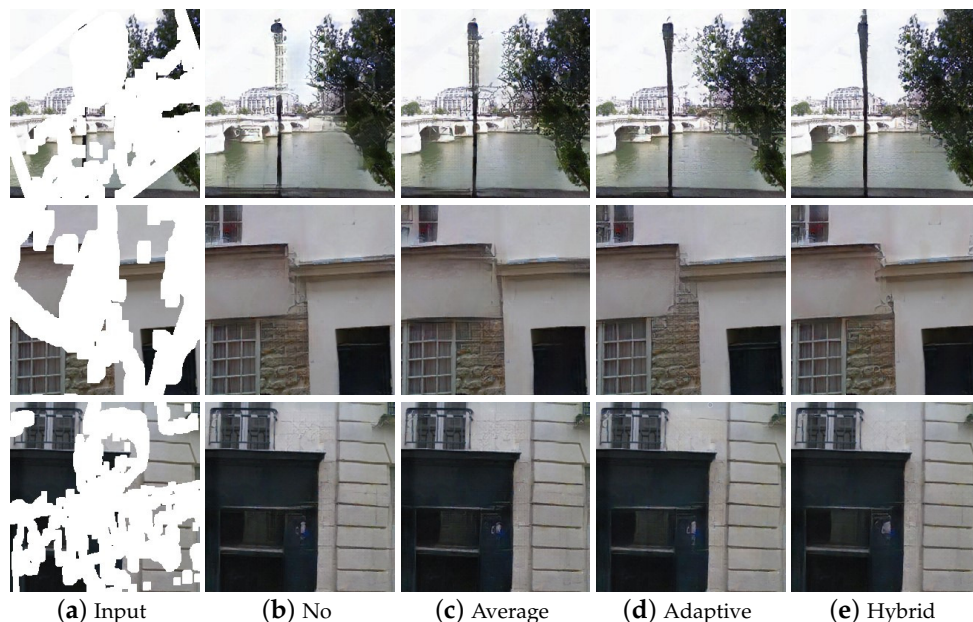


Figure 10. Different methods for feature merging. From the left to the right are: (a) Input, (b) No merging, (c) Average merging, (d) Adaptive merging, (e) Hybrid merging.

Point wise normalization. The results in the third row and fourth row demonstrate the performance with point wise normalization. Compared to the Batch Normalization counterpart, the PSNR and SSIM are significantly improved, this means that point-wise normalization is able to capture the discrepancy between valid and invalid pixels dur-

ing decoding processes, and learn the adaptive scale and bias parameters for assisting batch normalization.

The effect of iteration numbers As a hyper-parameter, the IterNums is set to seven in our model. In order to demonstrate the influence of different IterNums, we conduct the experiments on the Places2 dataset with large continuous holes. As shown in Table 3, the results from IterNums 5 are far from the others. This is because it is difficult to completely restore the large continuous holes in the small IterNums. Too many recursion times, resulting in far feature propagation, is also not conducive to image restoration.

Table 3. The influences of different IterNums.

IterNums	PSNR	SSIM	FID
5	18.39	0.605	40.21
7	19.53	0.669	34.27
9	19.32	0.661	35.93

5. Conclusions

In this paper, we propose a novel Multi-stage Feature Reasoning Generative Adversarial Network using recurrence filling to restore the arbitrary image defects. The hybrid weighted merging method fuses the feature map base to the mask characteristic and a learning process; it takes full advantage of the signals generated in every recurrence and thus eliminates gradient vanishing and increases the reasoning times of the model. Through the dynamic partial convolution, the image restoration range is adaptively adjusted according to the mask ratio. By this means, the correlation between hole boundary pixels and center area pixels is gradually strengthened, which is especially suitable for progressive image completion. Furthermore, skip-connection and point wise normalization are combined to minimize the loss of valid information in the up-sampling process; thus, the generated result structure is clearer and the content is more natural. Extensive experiments on the Places2, CelebA and Pairs StreetView datasets have demonstrated that MFR-GAN is more competitive than other methods in subjective quality and is objectively quantitative.

Author Contributions: Conceptualization, G.L. and L.L.; methodology, G.L.; software, G.L.; validation, G.L., L.L. and Y.P.; formal analysis, G.L. and L.L.; investigation, N.W.; resources, N.W. and X.Z.; data curation, N.W.; writing—original draft preparation, G.L.; writing—review and editing, L.L.; visualization, G.L.; supervision, L.L.; project administration, L.L.; funding acquisition, L.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by National Natural Science Foundation of China under grant No. 61573232, and was supported by Shaanxi Natural Science Foundation under grant No. 2022JM-335.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

- Shetty, R.R.; Fritz, M.; Schiele, B. Adversarial scene editing: Automatic object removal from weak supervision. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 7716–7726.
- Song, L.; Cao, J.; Song, L.; Hu, Y.; He, R. Geometry-aware face completion and editing. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 2506–2513.
- Armi, L.; Fekri-Ershad, S. Texture image analysis and texture classification methods—A review. *arXiv* **2019**, arXiv:1904.06554.
- Barnes, C.; Shechtman, E.; Finkelstein, A.; Goldman, D.B. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.* **2009**, *28*, 24. [[CrossRef](#)]

5. Ding, D.; Ram, S.; Rodríguez, J.J. Image inpainting using nonlocal texture matching and nonlinear filtering. *IEEE Trans. Image Process.* **2018**, *28*, 1705–1719. [[CrossRef](#)] [[PubMed](#)]
6. Lee, J.H.; Choi, I.; Kim, M.H. Laplacian patch-based image synthesis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2727–2735.
7. Criminisi, A.; Pérez, P.; Toyama, K. Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. Image Process.* **2004**, *13*, 1200–1212. [[CrossRef](#)] [[PubMed](#)]
8. Sagong, M.c.; Shin, Y.g.; Kim, S.w.; Park, S.; Ko, S.j. Pepsi: Fast image inpainting with parallel decoding network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–19 June 2019; pp. 11360–11368.
9. Nazeri, K.; Ng, E.; Joseph, T.; Qureshi, F.; Ebrahimi, M. Edgeconnect: Structure guided image inpainting using edge prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Long Beach, CA, USA, 15–19 June 2019.
10. Li, C.; He, K.; Liu, K.; Ma, X. Image Inpainting Using Two-Stage Loss Function and Global and Local Markovian Discriminators. *Sensors* **2020**, *20*, 6193. [[CrossRef](#)] [[PubMed](#)]
11. Yeh, R.A.; Chen, C.; Yian Lim, T.; Schwing, A.G.; Hasegawa-Johnson, M.; Do, M.N. Semantic image inpainting with deep generative models. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5485–5493.
12. Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; Huang, T.S. Free-form image inpainting with gated convolution. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 15–19 June 2019; pp. 4471–4480.
13. Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; Huang, T.S. Generative image inpainting with contextual attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5505–5514.
14. Zhang, H.; Hu, Z.; Luo, C.; Zuo, W.; Wang, M. Semantic image inpainting with progressive generative networks. In Proceedings of the 26th ACM international conference on Multimedia, Seoul, Korea, 22–26 October 2018; pp. 1939–1947.
15. Li, J.; Wang, N.; Zhang, L.; Du, B.; Tao, D. Recurrent feature reasoning for image inpainting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 7760–7768.
16. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1125–1134.
17. Darabi, S.; Shechtman, E.; Barnes, C.; Goldman, D.B.; Sen, P. Image melding: Combining inconsistent images using patch-based synthesis. *ACM Trans. Graph.* **2012**, *31*, 1–10. [[CrossRef](#)]
18. Hays, J.; Efros, A.A. Scene completion using millions of photographs. *ACM Trans. Graph.* **2007**, *26*, 4. [[CrossRef](#)]
19. Liu, H.; Jiang, B.; Xiao, Y.; Yang, C. Coherent semantic attention for image inpainting. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 15–19 June 2019; pp. 4170–4179.
20. Yi, Z.; Tang, Q.; Azizi, S.; Jang, D.; Xu, Z. Contextual residual aggregation for ultra high-resolution image inpainting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 7508–7517.
21. He, X.; Yin, Y. Non-Local and Multi-Scale Mechanisms for Image Inpainting. *Sensors* **2021**, *21*, 3281. [[CrossRef](#)] [[PubMed](#)]
22. Dong, H.; Liang, X.; Zhang, Y.; Zhang, X.; Shen, X.; Xie, Z.; Wu, B.; Yin, J. Fashion editing with adversarial parsing learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 8120–8128.
23. Liu, H.; Wan, Z.; Huang, W.; Song, Y.; Han, X.; Liao, J. PD-GAN: Probabilistic Diverse GAN for Image Inpainting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9371–9381.
24. Suvorov, R.; Logacheva, E.; Mashikhin, A.; Remizova, A.; Ashukha, A.; Silvestrov, A.; Kong, N.; Goka, H.; Park, K.; Lempitsky, V. Resolution-robust Large Mask Inpainting with Fourier Convolutions. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 4–8 January 2022; pp. 2149–2159.
25. Zhu, M.; He, D.; Li, X.; Li, C.; Li, F.; Liu, X.; Ding, E.; Zhang, Z. Image Inpainting by End-to-End Cascaded Refinement with Mask Awareness. *IEEE Trans. Image Process.* **2021**, *30*, 4855–4866. [[CrossRef](#)] [[PubMed](#)]
26. Yang, S.; Huang, R.; Han, F. Progressively Inpainting Images Based on a Forked-Then-Fused Decoder Network. *Sensors* **2021**, *21*, 6336. [[CrossRef](#)] [[PubMed](#)]
27. Pathak, D.; Krahenbuhl, P.; Donahue, J.; Darrell, T.; Efros, A.A. Context encoders: Feature learning by inpainting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2536–2544.
28. Zhao, L.; Mo, Q.; Lin, S.; Wang, Z.; Zuo, Z.; Chen, H.; Xing, W.; Lu, D. Uctgan: Diverse image inpainting based on unsupervised cross-space translation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 5741–5750.
29. Yan, Z.; Li, X.; Li, M.; Zuo, W.; Shan, S. Shift-net: Image inpainting via deep feature rearrangement. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 1–17.
30. Zeng, Y.; Fu, J.; Chao, H.; Guo, B. Learning pyramid-context encoder network for high-quality image inpainting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–19 June 2019; pp. 1486–1494.
31. Liu, G.; Reda, F.A.; Shih, K.J.; Wang, T.C.; Tao, A.; Catanzaro, B. Image inpainting for irregular holes using partial convolutions. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 85–100.
32. Iizuka, S.; Simo-Serra, E.; Ishikawa, H. Globally and locally consistent image completion. *ACM Trans. Graph.* **2017**, *36*, 1–14. [[CrossRef](#)]

33. Yu, T.; Guo, Z.; Jin, X.; Wu, S.; Chen, Z.; Li, W.; Zhang, Z.; Liu, S. Region normalization for image inpainting. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12733–12740.
34. Song, Y.; Yang, C.; Shen, Y.; Wang, P.; Huang, Q.; Kuo, C.C.J. Spg-net: Segmentation prediction and guidance network for image inpainting. *arXiv* **2018**, arXiv:1805.03356.
35. Xiong, W.; Yu, J.; Lin, Z.; Yang, J.; Lu, X.; Barnes, C.; Luo, J. Foreground-aware image inpainting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–19 June 2019; pp. 5840–5848.
36. Ren, Y.; Yu, X.; Zhang, R.; Li, T.H.; Liu, S.; Li, G. Structureflow: Image inpainting via structure-aware appearance flow. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 15–19 June 2019; pp. 181–190.
37. Li, J.; He, F.; Zhang, L.; Du, B.; Tao, D. Progressive reconstruction of visual structure for image inpainting. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 15–19 June 2019; pp. 5962–5971.
38. Guo, Z.; Chen, Z.; Yu, T.; Chen, J.; Liu, S. Progressive image inpainting with full-resolution residual network. In Proceedings of the 27th Acm International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 2496–2504.
39. Gatys, L.A.; Ecker, A.S.; Bethge, M. A neural algorithm of artistic style. *arXiv* **2015**, arXiv:1508.06576.
40. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
41. Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; Torralba, A. Places: A 10 million image database for scene recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 1452–1464. [[CrossRef](#)] [[PubMed](#)]
42. Liu, Z.; Luo, P.; Wang, X.; Tang, X. Deep learning face attributes in the wild. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3730–3738.
43. Doersch, C.; Singh, S.; Gupta, A.; Sivic, J.; Efros, A. What makes paris look like paris? *ACM Trans. Graph.* **2012**, *31*, 3. [[CrossRef](#)]
44. Xie, C.; Liu, S.; Li, C.; Cheng, M.M.; Zuo, W.; Liu, X.; Wen, S.; Ding, E. Image inpainting with learnable bidirectional attention maps. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 15–19 June 2019; pp. 8858–8867.