Masked Face Image Inpainting Based on Generative Adversarial Network

Qingyu Liu, Lei Chen, Yeguo Sun*, and Lei Liu

***Abstract*—Face image inpainting is a critical task in computer vision due to the intricate semantic and textural features of facial structures. While existing deep learning-based methods have achieved some progress, they often produce blurred or artifact- prone results when handling large occlusions, such as face masks. To address these challenges, this paper proposes a novel generative adversarial network (GAN) framework tailored for masked face inpainting. The generator adopts a U-Net architecture enhanced with a multi-scale mixed-attention residual module (MMRM), which integrates multi-branch convolutions for diverse receptive fields and combines spatial-channel attention mechanisms to prioritize semantically relevant features. The decoder further enhances feature fusion through channel attention mechanism, which selectively emphasizes meaningful patterns during feature map reconstruction. A realistic masked face dataset is synthesized using the CelebA database by dynamically adjusting mask positions, sizes, and angles based on facial landmarks, ensuring alignment with real-world scenarios. Quantitative and qualitative evaluations demonstrate that our method outperforms conventional models in both visual quality and quantitative metrics. Ablation studies further validate the effectiveness of MMRM and attention mechanisms in preserving structural coherence and reducing artifacts.**

***Index Terms*—Attention mechanism, Generative adversarial network, image inpainting, residual module**

I. INTRODUCTION

Face recognition, a type of biometric technology, has been widely used in different areas of social life, and some excellent face recognition technologies have even surpassed the level of human recognition. However, the current face recognition accuracy is still affected by other external factors, such as mask blocking. When the face is obscured by a mask, the effective features that can be extracted from the face are greatly reduced, which leads to a lower accuracy rate of face recognition. Image inpainting is the process of inferring and reconstructing the content of a missing area from limited known image information, making the image visually complete and natural. Using image inpainting techniques to remove face occlusion and try to restore the face to its initial state has become a hot issue of concern. This research presents a new generative adversarial network (GAN) model for face inpainting.

Image inpainting techniques, in terms of their development process, have two main categories: traditional techniques and deep learning-based techniques. The traditional image inpainting techniques are mainly diffusion-based [1], [2] and block-based methods [3], [4]. The primary principle of the diffusion inpainting algorithms is to disperse the pixels from the image's unobstructed regions to the obstructed regions to achieve the goal of inpainting. Bertalmio et al. [5] were the first to apply the method to image inpainting by proposing the BSCB model, which used the heat diffusion equation in fluid dynamics to propagate domain information into the occluded regions. Efros and Leung et al. [6] proposed a Markov random field-based approach to fill the occluded regions by selecting the best matching sample blocks from the unoccluded regions. In order to reduce the cost of searching and matching, the PatchMatch approach was presented by Barnes et al. [7] and used a quick nearest neighbor algorithm for searching. Such methods have an assumption that the sample image contains similar information about the occluded regions. This is not guaranteed in many cases. In addition, the block-based inpainting methods rely on a large training dataset and are too computationally intensive.

In 1998, the LeNet model, one of the first convolutional neural network (CNN) models, was introduced by LeCun et al. [8]. This model catalyzed the advancement of CNN. Image inpainting techniques based on deep learning have been rapidly developed since the advent of CNN. The methods of deep learning [9], [10], [11], [12] can effectively extract the features of the images and fill the occluded areas of the images, to achieve the purpose of inpainting. In 2014, Goodfellow et al. [13] pioneered GAN. GAN pushed image processing to a more advanced stage of development. In the model, the discriminator and generator are trained against one another, and the two improve their respective abilities during the training, as shown in Figure 1. Using the trained generator we can fit real data distributions to achieve tasks such as image generation and image inpainting. In the year 2016, Pathak et al. [14] first used GAN for image inpainting by proposing the Context Encoder model. This model employs an encoder-decoder network topology, in which the encoder is used for image feature extraction and the decoder creates an image the

The work is supported in part by the Key Projects of Natural Science Research in Anhui Colleges and Universities under Grant 2023AH051546, in part by the University Natural Science Foundation of Anhui Province under Grant 2022AH010085, in part by the Opening Foundation of State Key Laboratory of Cognitive Intelligence under Grant COGOS-2023HE02, in part by Program of Anhui Education Department under Grant 2024jsqygz83.

Qingyu Liu, Lei Chen and Lei Liu are with Faculty of School of Computer Science, Huainan Normal University, Huainan, China (e-mail: liuqy@hnnu.edu.cn, LeiChn0911@163.com, 357920185@qq.com).

*Yeguo Sun is with Faculty of School of Finance and Mathematics, Huainan Normal University, Huainan, China (*Corresponding Author, e-mail: yeguosun@126.com).

DOI: 10.36244/ICJ.2025.2.2

same size as the obscured image. Regularly obscured images can be restored using the Context Encoder model, although the restored images have issues including blurring and inconsistent content. Subsequently, Iizuka et al. [15] proposed a bi-discriminators GAN model. The generated images and real images are fed into the global discriminator and the images are evaluated as a whole for consistency to ensure the overall coherence of the images. The use of local and global discriminators enables the generator to better learn the data distribution of real images, thereby improving the model's restoration capabilities. Yu et al. [16] introduced a new end-to-end GAN for image inpainting using coarse and fine inpainting networks in tandem. The coarse inpainting stage accomplishes the initial contour generation of the occluded image, and the fine restoration realizes the image fine processing on this basis, which ultimately results in a high-quality inpainting result.

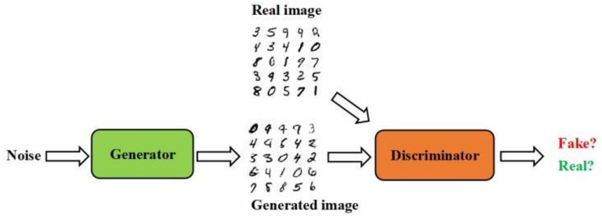

Fig. 1. GAN structure.

Face image inpainting is a sub-task of image inpainting, but it is special due to the complex texture structure and rich semantic features of the face [17], [18], [19]. GAN were initially used in the field of face image inpainting by Li et al. [20]. He proposed the concept of semantic parsing loss in the loss functions and adopted a pre-trained parsing network for face image inpainting. Nazeri et al. [21] introduced a 2-step inpainting method in which the first step predicts and restores the contours of the face and then restores the detailed aspects of the face. Qin et al. [22] proposed a prescription based on weighted face similarity for restoring face images with relatively large occluded regions. Yang et al. [23] used a bi-discriminator to restore occluded face images to maintain better semantic consistency. Although these improved face image inpainting methods based on GAN can perform face image inpainting, there are still some problems. For example, how to ensure the balanced training of the generator and discriminator, how to realize the fine inpainting of face images and reduce the complexity of the model, how to avoid using extra input information such as edge contours in the face inpainting process, and so on. To address these issues, we proposed a GAN model based on multi-scale mixed-attentions residual module (MMRM). The following are the primary contributions of this work:

(a) Based on the public CelebA dataset, the masks are added by accurately detecting the positions of the face feature points. The position, size and angle of the face in each image are different, and thus the position, size and angle of the face mask will also change. Therefore, the produced dataset is more in line with the actual situation of people wearing masks, and the trained generator model has stronger generalization ability.

(b) The generator model is based on the U-Net structure [24], and the feature fusion of down-sampled texture features and up-sampled high-level semantic features can be performed through skip-layer connections to improve the image inpainting of faces.

(c) We use different scales of convolutional branches for the residual module. Each branch uses spatial attention [25] to extract features, and multiple branch features are fused before adding channel attention [26] to focus on more effective channel features.

II. Proposed Method

GAN was originally proposed in 2014 and the original inputs to the generator are random vectors. We use GAN for face image inpainting, so the inputs to the generator are the face images wearing masks and the outputs are the restored face images without masks. Figure 2 displays the model's general architecture. The generator in the model makes use of the U-Net structure, and the discriminator is also part of the model. U-Net has the capacity to extract features at various scales in the encoding phase and fuse the extracted features in the decoding phase. This network can extract face semantic information quickly and makes training easier and convergence faster. The discriminator adopts the PatchGAN [27] structure and uses a spectrally normalized convolutional layer instead of the normal convolutional layer, which improves the texture of the restored region and has the effect of stabilizing the training. The generator uses MMRM for both the modules of up-samplings and down-samplings.

A. Generator

Because of its skip connections, the U-Net structure was initially used in the field of medical image segmentation, preserving both high-level semantic information and low-level characteristics. Thus, we combine the U-Net structure with GAG. We improve the original residual module by setting it to three convolutional branches, each of which uses a convolutional kernel of different sizes to extract features separately. Meanwhile, each branch uses a spatial attention module to enhance feature extraction. Figure 3 shows the structure of the improved residual module.

Parameter updates in CNN rely on backpropagation of the gradient. As the depth of the network gradually increases, parameters less than 1 are multiplied together resulting in gradient vanishing. When the network weights are initialized to a larger value, the gradient grows exponentially as the neural network deepens, triggering gradient explosion. Although the gradient problem can be solved to a certain extent by introducing activation functions such as ReLU and Batch Normalization layer, when the depth of the network is increased, it still occurs that the training effect of the deeper network is rather worse than that of the shallower network, i.e., the network degradation problem. He et al. [28] proposed ResNet network in 2016 for solving the degradation problem that occurs when the network depth increases. Thus, both the up-sampling modules and down-sampling modules of the generator use improved residual modules.

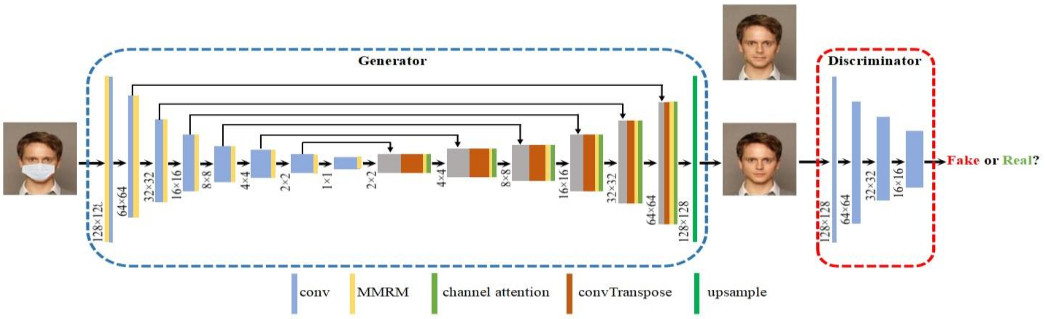

Fig. 2. Architecture of the model.

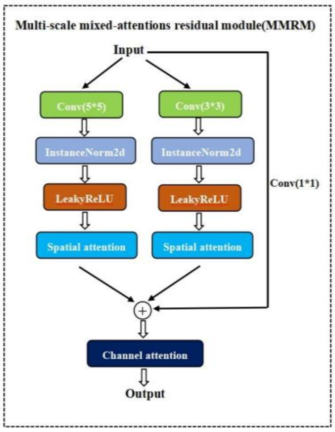

Fig. 3. Architecture of MMRM.

The improved residual module comprises three parallel convolutional branches. The convolution kernel sizes of the three convolution branches are 1×1, 3×3, 5×5, and they have different receptive fields so that features of different scales can be extracted. Spatial attention is introduced in 3×3 and 5×5 branches to dynamically weight the spatial location of the feature map. For example, when restoring mask-obscured regions, the model prioritizes attention to unobscured semantic key points (e.g., bridge of the nose) to enhance the guidance of contextual information. After multi-branch feature splicing, important channel features are filtered by channel attention. For example, channels corresponding to high-frequency information such as facial skin color and hair texture are enhanced, while redundant background channels are suppressed to improve the semantic consistency of the inpainting.

Figure 3 also shows that the spatial attention layers are added to the 3×3 and 5×5 convolutional branches, respectively. The goal of face inpainting is to fill the occluded region with information from the non-occluded regions. Considering that the face image has obvious geometric features, it is obvious that different regions of the face have different effects on the final inpainting result. The final output of the residual module is directly related to the input. Additionally, it is necessary to consider whether the weights of each convolutional branch should be identical. Thus, three convolutional branches use channel attention after the channel dimension is concatenated, focusing on important information to suppress irrelevant information.

B. Discriminator

In the image inpainting task, the most important thing is the prediction of the pixel values of the masked region. In adversarial learning, a discriminator constrains the training of the generator by determining whether the images are similar to the real. A good discriminator maintains the overall semantic consistency of the restored images and enhances the texture details of the restored regions. In Figure 4, the network structure is depicted and we employ PatchGAN as the discriminator structure. The discriminator network consists of a stack of 4 convolutional layers with convolutional kernel size 3*3 and stride size 2. The purpose is to obtain the feature statistics of Markov Patch (MP). Each point in the output matrix represents a region of the original input image, by which the efficiency of the discriminator can be improved.

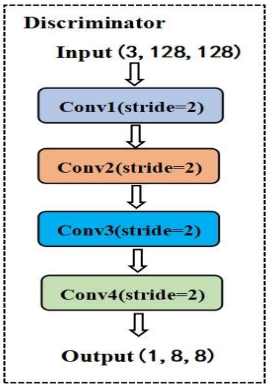

Fig. 4. Discriminator.

C. Loss functions

The discriminator and generator are trained adversarially using loss functions. The generator and discriminator iteratively improve their respective capabilities until reaching a steady state. Therefore, loss functions are crucial for GAN. To make the generator better at fixing occluded face images, the generator loss functions use adversarial loss, L1 loss, TV Loss, and they are defined as below:

$$Loss_{adv} = -E_{z\sim P_z} D(G(z)) \tag{1}$$

$$Loss_{L1} = E_{z\sim P_z,\ x\sim P_{data}} \parallel G(z) - x \parallel \tag{2}$$

where z denotes the masked face image and x denotes the real face image. D and G represent the generator and discriminator respectively. Considering the coherence of the restored images, we also used the TV loss:

$$Loss_{TV} = \sum_{i,j} \sqrt{(I_{i,j-1} - I_{i,j})^2 + (I_{i+1,j} - I_{i,j})^2} \qquad (3)$$

where $I_{i,j}$ denotes a pixel point of the image, and $I_{i,j-1}$, $I_{i+1,j}$ are the neighboring pixel points in the vertical and horizontal directions, respectively. Thus, the generator's loss function is expressed as follows:

$$Loss(G) = \gamma_{adv} * Loss_{adv} + \gamma_{L1} * Loss_{L1} + \gamma_{TV} * Loss_{TV} \quad (4)$$

Where $Loss_{adv}$ represents the adversarial loss function, $Loss_{L1}$ represents the L1 loss function, and $Loss_{TV}$ represents the TV Loss function. The weights of the different loss functions are shown by the symbols γ_{adv} , γ_{L1} , and γ_{TV} , respectively.

The discriminator then has only the adversarial loss function, as follows:

$$\mathrm{Loss(D)} = Loss_{adv} \qquad (5)$$

III. EXPERIMENTS AND RESULTS

In this section, we produced a dataset of faces wearing masks. Then, quantitative and qualitative experiments are conducted to compare with other end-to-end image inpainting models such as Pix2Pix [29]. Finally, to show how effective the improved model is, ablation experiments are carried out.

A. Dataset

An excellent dataset for face processing is the CelebA image dataset, an open dataset created by the Chinese University of Hong Kong that includes 202599 images of over 10000 individuals from all over the world. In this study, 40000 of these images were chosen at random to make up the initial dataset, and the face recognition library "Face_recognition" is used to accurately locate the face feature points to complete the addition of masks. Considering the hardware limitations, all images are scaled to 128×128 size uniformly.

The "Face_recognition" library could detect faces in an image and obtain the locations of the key points such as eyes, nose, mouth, etc., as shown in Figure 5. To realistically produce a dataset that matches the real situation, the mask should be able to cover the mouth, chin, and tip of the nose of the face. Also, considering the different positions and angles of the face in the image, the angle and size of the face mask should be changed accordingly. This is extremely important. The size, position and angle of each face may vary. We can adaptively add masks based on the key points of the specific face image. Thus, the produced dataset has better diversity, and the trained model has better adaptability. The specific process of adding a mask to a face is shown in Algorithm 1.

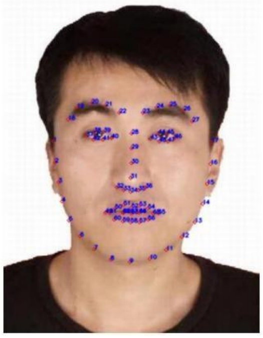

Fig. 5. Key points of the face. There are a total of 68 feature points, each of which is numerically numbered, and the coordinates of each feature point can be obtained.

```
Algorithm 1 Steps to Produce Masked Face Images
Input: face image without mask
Output: masked face image
1 Detect faces and get all the key points' coordinates;
2 Calculate the distance between point 29 of the nose and point 8 of the chin, and then multiply by 1.2 to get the height of the new mask, denoted as H;
3 Calculate the distance between point 29 of the nose and point 1 of the right side of the face, and then multiply by 1.2 to get the width of the right half of the new mask, denoted as $W_R$;
4 Intercept the right half of the original mask and transform the size according to $(H, W_R)$;
5 Calculate the distance between point 29 of the nose and point 16 on the left side of the face and multiply by 1.2 to get the width of the left half of the new mask, denoted $W_L$;
6 Intercept the left half of the original mask and transform to $(H, W_R)$;
7 Merge the left and right sides of the new mask to get a completely new mask;
8 Place the new mask using the center node between point 29 of the nose and point 8 of the chin as a reference point;
9 Determine if nose point 33 and chin point 8 are perpendicular, if not, do the same angular transformation using the geometric center point of the new mask as a datum.
```

In the above process, the height and width of the new face mask are multiplied by 1.2 from the original calculated values. This is because the mask is meant to cover the key locations such as the nose and mouth, and the values should be slightly larger than the calculated distance. This produces a more realistic image. Figure 6 shows the complete process of adding the mask, following the same steps as above. The input to the algorithm is a normal face, followed by the generation of the right half of the mask and the left half of the mask. Then later on it is spliced to form a complete mask and covered in the right place of the face. The last step is to adjust the angle of the mask. Following this method, we choose the three most common masks in daily life to produce the dataset images, as shown in Figure 7.

B. Experimental environment and parameter settings

The experimental hardware environment is an Intel Xeon processor, with 128GB RAM and 8 NVIDIA RTX 3090 Ti GPU cards. The software environment uses Red Hat 4.8.5, Python 3.7.1, and PyTorch 1.10. The model is trained using data parallelism, in which the dataset images are evenly distributed across all computational cards. During the training process, each node updates its model parameters independently, and finally, the model parameters of each node are fused to obtain the final model. In this way, the computation and memory consumption of a single node can be greatly reduced, and the overall training speed of the model can be improved.

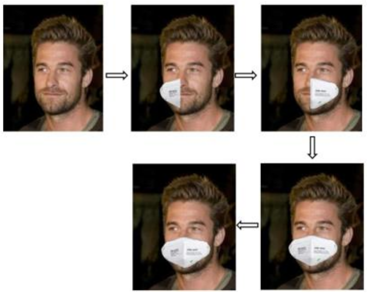

Fig. 6. Mask addition process.

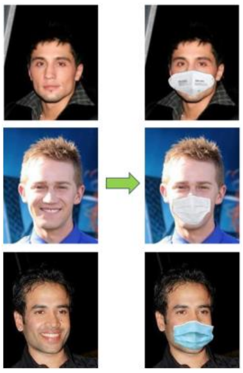

Fig. 7. Masked images with three types of face masks.

The experimental hyperparameters are set in Table I. The weights of the generator adversarial loss, L1 loss, and TV loss are 0.01, 0.95, and 0.04, respectively. The model optimizer uses Adam with an initial learning rate of 0.0002, and the batch size is set to 1500 according to the GPU memory capacity. For later comparison experiments, all the image inpainting models are trained for a maximum of 1000 epochs.

The model took 19 hours for a single training process, and used about 23GB of each GPU's memory (for a total of 24GB on a single card).

TABLE I
TRAINING HYPERPARAMETERS

Parameter	Value	Parameter	Value
γ_{adv}	0.01	Learning rate	0.0002
γ_{L1}	0.95	Epochs	1000
γ_{TV}	0.04	Optimizer	Adam
Batch size	1500		

C. Experimental results

In this section, through some contrast experiments, we qualitatively and quantitatively evaluate the results of our proposed method. The models we use as comparisons are Pix2Pix [30], Shift-Net [31], GMCNN [32], WaveFill [33], etc.

1) Qualitative analysis

Figure 8 displays the comparison results using our new dataset. The first columns are the initial face images without masks and the second columns are the images after the masks are automatically added according to Algorithm 1. The next five columns show the comparison results of different inpainting methods. Pix2Pix uses the U-Net structure to reconstruct face images, but it doesn't work well for face images with complex image structures. So we can see obvious blurred pixels. Shift-Net adds a shift-connection layer to the generator network based on the Pix2Pix model and adds guidance loss to the overall loss functions, so it gives better inpainting results than Pix2Pix. The restored regions (e.g., nose and mouth) exhibit significantly better quality compared to Pix2Pix. GMCNN is still the architecture for GAN, but its generators use a multi-scale CNN structure. In terms of feature extraction, the use of a multicolumn structure allows the image to be decomposed into components with different receptive fields, synthesizing features at different scales. The key problem with GMCNN is that it does not apply to large-scale datasets. Using a wavelet-based inpainting technique, WaveFill divides images into three frequency bands and fills in the areas that are absent in each band independently. This method effectively mitigates the conflict between low and high frequencies. That's why it's better in painting than the previous 3 methods, e.g., the complex structure of the mouth visually looks more coherent. In contrast, within 1000 epochs with more accurate local texture information, our model can reconstruct symmetrical, full-face images. In rows 1-3, for instance, the mouth texture is distinct and symmetrical.

To further verify that our proposed model can restore semantically informative face occlusion images, we also try to add rectangular occlusions to the face images, where the length and width of the occlusion region are 1/2 of the length and width of the images, respectively. The experimental results of the comparison are shown in Figure 9. The inpainting effect of our improved model is better than the other four methods. Compared with Figure 8, although the area of the rectangular occlusion region is larger than the face mask occlusion region, we find that the rectangular occlusion image is restored with much less blurring and artifacts. This is because the rectangular occlusion is structurally symmetric and fixed in size, whereas the face masks have different sizes, traits, and angles as they are added to the faces. The inpainting results are visually more coherent and natural. But compared to the original images in column (a), the larger areas of occlusions lead to greater diversity, the restored images are less similar to the original images.

2) Quantitative analysis

To objectively compare the inpainting results of the improved model with other methods, the inpainting outcomes are statistically evaluated in this paper using three indicators: peak signal-to-noise ratio (PSNR) [34], structural similarity (SSIM) [35] and learned perceptual image patch similarity (LPIPS) [36]. PSNR assesses the restored effect by comparing the two samples' pixel values. SSIM analyzes the variations in brightness, contrast, and structure. LPIPS is a deep learning-based image similarity assessment metric that evaluates the similarity of two images by comparing their local perceptual features. Within the realm of image inpainting, they are three of the most commonly used evaluation metrics.

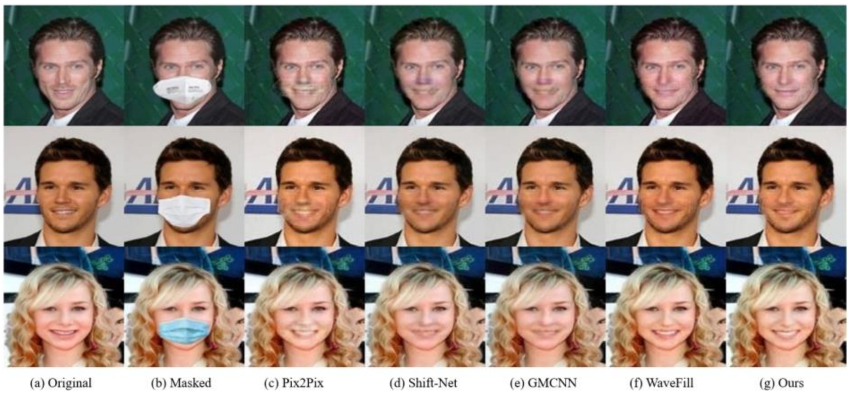

Fig. 8. Visual comparison of our model with other models on the face mask occlusion dataset. From left to right: (a) original images without masks, (b) input masked images, (c) Pix2Pix, (d) Shift-Net, (e) GMCNN, (f) WaveFill, (g) outputs of our model.

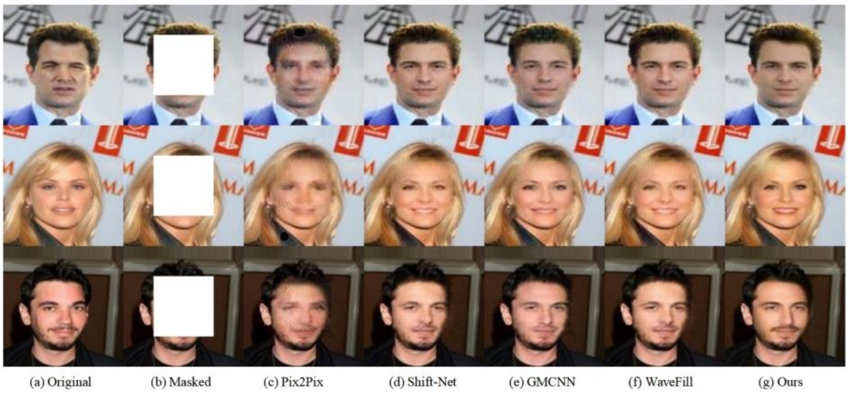

Fig. 9. Visual comparison of our model with other models on the rectangular occlusion dataset. From left to right: (a) original images without masks, (b) input masked images, (c) Pix2Pix, (d) Shift-Net, (e) GMCNN, (f) WaveFill, (g) outputs of our model.

The quantitative comparison results for the face mask occlusion dataset are shown in Table II. For image inpainting, the larger their value, the better the inpainting result is. Table III shows comparisons of quantitative results on the rectangular occlusion dataset. Experimental results display that our proposed face inpainting model is superior to several other methods. We also find that the experimental results of the rectangular occlusion face inpainting are numerically worse than the mask occlusion face. This is also in line with our previous analysis, as the rectangular occlusion covers most of the face area. The restored image is enough to ensure the coherence and naturalness of the restored face image. While PSNR, SSIM and LPIPS mainly evaluate the similarity between the restored image and the original image, Table III is numerically lower than Table II.

TABLE II
COMPARISONS OF QUANTITATIVE RESULTS ON THE FACE MASK OCCLUSION DATASET

Method	PSNR	SSIM	LPIPS
Pix2Pix	28.74	90.21	9.51
Shift-Net	29.78	91.12	9.14
GMCNN	29.57	90.61	9.15
WaveFill	30.19	91.24	8.93
Ours	30.75	91.96	8.13

TABLE III
COMPARISONS OF QUANTITATIVE RESULTS ON THE RECTANGULAR OCCLUSION DATASET

Method	PSNR	SSIM	LPIPS
Pix2Pix	24.99	85.83	13.32
Shift-Net	25.34	86.16	13.14
GMCNN	24.40	83.11	16.59
WaveFill	25.41	86.72	12.43
Ours	26.39	88.24	11.00

3) Ablation experiments

To verify the validity of our model structure, we performed ablation experiments. In the complete generator network model, spatial attention in MMRM, channel attention in MMRM, the entire MMRM module, and spatial attention in up-sampling are removed, respectively. The ablation experiments are then performed on the face mask occlusion dataset and the rectangular occlusion dataset, as shown in Figure 10 and Figure 11.

MMRM can effectively perceive global structural and semantic information to reconstruct complete edge information and improve edge distribution in occluded regions. For example, in Figure 10(e) and Figure 11(e), there are prominent boundaries and blurring. The generator model utilizes the convolutional branches in the MMRM module to extract the salient information in space and channel. The consistency of

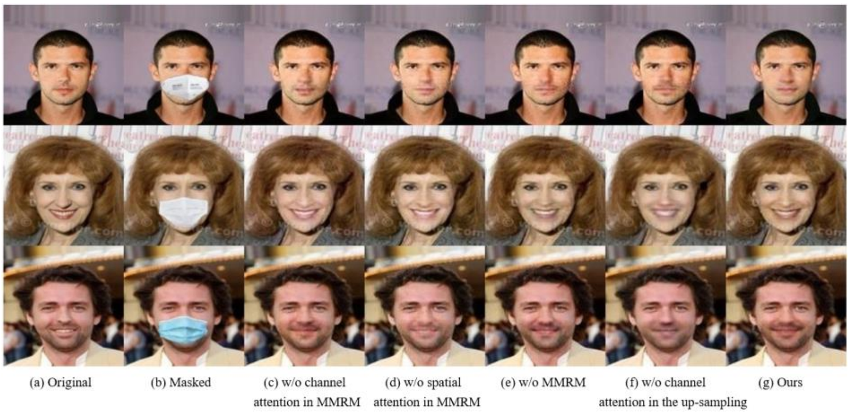
(a) Original
(b) Masked
(c) w/o channel attention in MMRM
(d) w/o spatial attention in MMRM
(e) w/o MMRM
(f) w/o channel attention in the up-sampling
(g) Ours

Fig. 10. Visual comparison of ablation experiments on the face mask occlusion dataset. From left to right: (a) original images without masks, (b) input masked images, (c) without channel attention in MMRM, (d) without spatial attention in MMRM, (e) without MMRM, (f) without channel attention in the up-sampling, (g) outputs of our model.

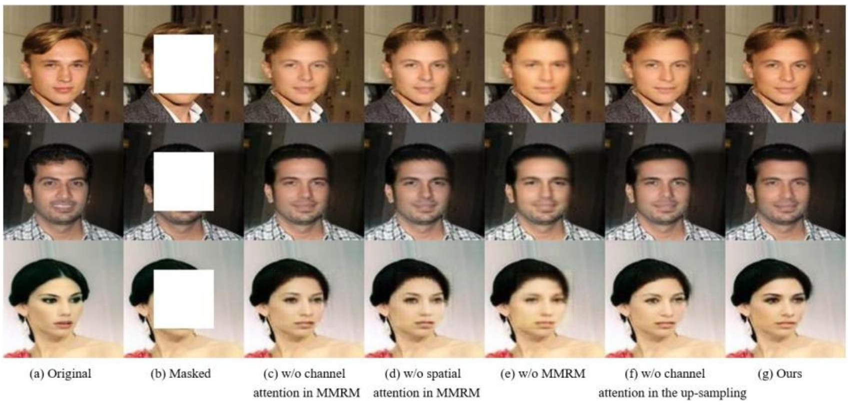
(a) Original
(b) Masked
(c) w/o channel attention in MMRM
(d) w/o spatial attention in MMRM
(e) w/o MMRM
(f) w/o channel attention in the up-sampling
(g) Ours

Fig. 11. Visual comparison of ablation experiments on the rectangular occlusion dataset. From left to right: (a) original images without masks, (b) input masked images, (c) without channel attention in MMRM, (d) without spatial attention in MMRM, (e) without MMRM, (f) without channel attention in the up-sampling, (g) outputs of our model.

convolutional branches in the MMRM module to extract the salient information in space and channel. The consistency of boundary information requires the gradual fusion of different spatial and channel semantics, which ultimately leads to the accurate reconstruction of global contours and local structures. Comparing columns (c) and (d), we find that if the spatial attention in MMRM is removed, there is significant blurring in the restored face image. For face inpainting, not all regions are equally important for occluded regions. With the loss functions, the spatial attention of the generator network when performing gradient backpropagation is looking for the most important parts for feature extraction. The feature maps acquired from the generator's up-sampling modules are concatenated, or superimposed at the channel level, with the left feature maps. Thus we use channel attention to focus on the key information. So, in Figure 10 (f) and Figure 11 (f), there are blurring of structurally complex parts such as the mouth and eyes.

Tables IV and V present the quantitative results of the ablation experiments. The results of the experiments on the face mask occlusion dataset and the rectangular occlusion dataset remain generally consistent with the results of the qualitative analysis above. The MMRM module in the generator model is the most useful, and once removed, the experimental data declines the most on both of the 2 datasets. The other three columns also demonstrate the role of spatial and channel attention in face image inpainting, and their removal brings about a decrease in both metrics.

TABLE IV
COMPARISONS OF QUANTITATIVE RESULTS OF ABLATION EXPERIMENTS ON THE FACE MASK OCCLUSION DATASET

	w/o channel attention in MMRM	w/o spatial attention in MMRM	w/o MMRM	w/o channel attention in the up-sampling
PSNR	29.19	29.16	28.43	29.15
SSIM	91.45	91.18	90.32	91.35
LPIPS	8.37	8.61	8.75	8.44

TABLE V
COMPARISONS OF QUANTITATIVE RESULTS OF ABLATION EXPERIMENTS ON THE RECTANGULAR OCCLUSION DATASET

	w/o channel attention in MMRM	w/o spatial attention in MMRM	w/o MMRM	w/o channel attention in the up-sampling
PSNR	25.86	25.12	23.45	25.74
SSIM	85.12	84.48	82.31	86.56
LPIPS	12.37	12.81	13.48	13.24

4) Cross-datasets experiments

In addition, we conducted across datasets experiments. The adopted dataset, CelebA-HQ, which is a high-quality face image designed specifically for computer vision research, is different from CelebA. Visual comparison results and quantitative comparisons are shown in Figure 12 and Table Ⅵ. The experimental results show that the inpainting effect of our design model is better than the results of several other models.

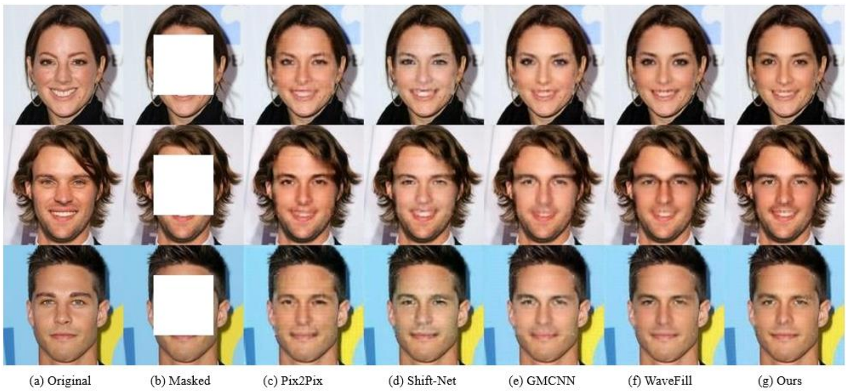

Fig. 12. Visual comparison of our model with other models on CelebA-HQ dataset. From left to right: (a) original images without masks, (b) input masked images, (c) Pix2Pix, (d) Shift-Net, (e) GMCNN, (f) WaveFill, (g) outputs of our model.

TABLE Ⅵ
COMPARISONS OF QUANTITATIVE RESULTS ON CELEBA-HQ DATASET

Method	PSNR	SSIM	LPIPS
Pix2Pix	25.28	86.26	9.78
Shift-Net	26.47	87.61	9.43
GMCNN	26.62	87.76	9.57
WaveFill	26.88	89.26	8.68
Ours	27.87	89.81	7.42

IV. CONCLUSION

We present a generative adversarial network model for face image inpainting in this paper. The discriminator network and the generator network are the two primary components of the model. The U-Net topology forms the foundation of the generator network and MMRM is used in the sampling module. This module fuses two attentional mechanisms into the residual branch, improving the network's ability to perceive global structural features and ameliorating the problem of inconsistent image inpainting boundaries. We reconstructed the dataset using the face feature point detection method and experiments demonstrate that the model performs well when reconstructing complex structures of faces. In future work, we will concentrate on reducing computational time, training the network with fewer parameters, and optimizing the architecture.

REFERENCES

[1] H. Li, W. Luo, and J. Huang, “Localization of diffusion-based inpainting in digital images,” *IEEE transactions on information forensics and security*, vol. 12, no. 12, pp. 3050–3064, 2017. **DOI**: 10.1109/TIFS.2017.2730822.

[2] K. Li, Y. Wei, Z. Yang, and W. Wei, “Image inpainting algorithm based on TV model and evolutionary algorithm,” *Soft Computing*, vol. 20, pp. 885–893, 2016. **DOI**: 10.1007/s00500-014-1547-7.

[3] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen, “Image melding: Combining inconsistent images using patch-based synthesis,” *ACM Transactions on graphics (TOG)*, vol. 31, no. 4, pp. 1–10, 2012. **DOI**: 10.1145/2185520.2185578.

[4] T. Ružić and A. Pižurica,“Context-aware patch-based image inpaintingˇ using markov random field modeling,” *IEEE transactions on image processing*, vol. 24, no. 1, pp. 444–456, 2014. **DOI**: 10.1109/TIP.2014.2372479.

[5] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, “Image inpainting,” pp. 417–424, 2000. **DOI**: 10.1145/344779.344972.

[6] A. A. Efros and T. K. Leung, “Texture synthesis by non-parametric sampling,” in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2. IEEE, 1999, pp. 1033–1038. **DOI**: 10.1109/ICCV.1999.790383.

[7] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, “Patchmatch: A randomized correspondence algorithm for structural image editing,” *ACM Trans. Graph.*, vol. 28, no. 3, p. 24, 2009. **DOI**: 10.1145/1531326.1531330.

[8] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” in *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. **DOI**: 10.1109/5.726791.

[9] Z. Qin, Q. Zeng, Y. Zong, and F. Xu, “Image inpainting based on deep learning: A review,” *Displays*, vol. 69, p. 102 028, 2021. **DOI**: 10.1016/j.displa.2021.102028.

[10] J. Jam, C. Kendrick, K. Walker, V. Drouard, J. G.-S. Hsu, and M. H. Yap, “A comprehensive review of past and present image inpainting methods,” *Computer vision and image understanding*, vol. 203, p. 103 147, 2021. **DOI**: 10.1016/j.cviu.2020.103147.

[11] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, and Y. Akbari, “Image inpainting: A review,” *Neural Processing Letters*, vol. 51, pp. 2007–2028, 2020. **DOI**: 10.48550/arXiv.1909. 06399.

[12] D. J. B. Rojas, B. J. T. Fernandes, and S. M. M. Fernandes, “A review on image inpainting techniques and datasets,” in *2020 33rd SIBGRAPI conference on graphics, patterns and images (SIBGRAPI)*. IEEE, 2020. pp. 240–247. **DOI**: 10.1109/SIBGRAPI51738.2020.00040.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” *Advances in neural information processing systems*, vol. 27, 2014. **DOI**: 10.48550/arXiv.1406.2661.

[14] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, “Context encoders: Feature learning by inpainting,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2536–2544. **DOI**: 10.48550/arXiv.1604.07379.

[15] S. Iizuka, E. Simo-Serra, and H. Ishikawa, “Globally and locally consistent image completion,” *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, pp. 1–14, 2017. **DOI**: 10.1145/3072959.3073659.

[16] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5505–5514. **DOI**: 10.48550/arXiv.1801.07892.

[17] C. Han and J. Wang, "Face image inpainting with evolutionary generators," *IEEE Signal Processing Letters*, vol. 28, pp. 190–193, 2021. **DOI**: 10.1109/LSP.2020.3048608.

[18] X. Zhang, X. Wang, C. Shi, Z. Yan, X. Li, B. Kong, S. Lyu, B. Zhu, J. Lv, Y. Yin et al., "De-gan: Domain embedded gan for high quality face image inpainting," *Pattern Recognition*, vol. 124, p. 108 415, 2022. **DOI**: 10.48550/arXiv.2002.02909.

[19] Y. Dogan and H. Y. Keles, "Iterative facial image inpainting based on an encoder-generator architecture," *Neural Computing and Applications*, vol. 34, no. 12, pp. 10 001–10 021, 2022. **DOI**: 10.48550/arXiv.2101.07036.

[20] Y. Li, S. Liu, J. Yang, and M.-H. Yang, "Generative face completion," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3911–3919. **DOI**: 10.48550/arXiv.1704.05838.

[21] K. Nazeri, "Edgeconnect: Generative image inpainting with adversarial edge learning," arXiv preprint arXiv:1901.00212, 2019. **DOI**: 10.48550/arXiv.1901.00212.

[22] J. Qin, H. Bai, and Y. Zhao, "Face inpainting network for large missing regions based on weighted facial similarity," *Neurocomputing*, vol. 386, pp. 54–62, 2020. **DOI**: 10.1016/j.neucom.2019.12.079.

[23] X. Yang, P. Xu, Y. Xue, and H. Jin, "Contextual feature constrained semantic face completion with paired discriminator," *IEEE Access*, vol. 9, pp. 42 100–42 110, 2021. **DOI**: 10.1109/ACCESS.2021.3065661.

[24] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference*, Munich, Germany, October 5-9, 2015, proceedings, part III 18. Springer, 2015, pp. 234–241. **DOI**: 10.48550/arXiv.1505.04597.

[25] M. Zhu, L. Jiao, F. Liu, S. Yang, and J. Wang, "Residual spectral–spatial attention network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 1, pp. 449–462, 2020. **DOI**: 10.1109/TGRS.2020.2994057.

[26] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 534–11 542. **DOI**: 10.48550/arXiv.1910.03151.

[27] G. Chen, G. Zhang, Z. Yang, and W. Liu, "Multi-scale patch-GAN with edge detection for image inpainting," *Applied Intelligence*, vol. 53, no. 4, pp. 3917–3932, 2023. **DOI**: 10.1007/s10489-022-03577-2.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. **DOI**: 10.1109/CVPR.2016.90.

[29] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134. **DOI**: 10.48550/arXiv.1611.07004.

[30] Z. Yan, X. Li, M. Li, W. Zuo, and S. Shan, "Shift-net: Image inpainting via deep feature rearrangement," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 1–17. **DOI**: 10.48550/arXiv.1801.09392.

[31] Yan Z, Li X, Li M, et al, "Shift-net: Image inpainting via deep feature rearrangement", in *Proceedings of the European conference on computer vision (ECCV)*. 2018: 1-17. **DOI**: 10.1007/978-3-030-01264-9_1.

[32] Y. Wang, X. Tao, X. Qi, X. Shen, and J. Jia, "Image inpainting via generative multi-column convolutional neural networks," *Advances in neural information processing systems*, vol. 31, 2018. **DOI**: 10.48550/arXiv.1810.08771.

[33] Y. Yu, F. Zhan, S. Lu, J. Pan, F. Ma, X. Xie, and C. Miao, "Wavefill: A wavelet-based generation network for image inpainting," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 14 114–14 123. **DOI**: 10.48550/arXiv.2107.11027.

[34] A. Dziembowski, D. Mieloch, J. Stankowski, and A. Grzelka, "IV-PSNR- the objective quality metric for immersive video applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 11, pp. 7575–7591, 2022. **DOI**: 10.1109/TCSVT.2022.3179575.

[35] I. Bakurov, M. Buzzelli, R. Schettini, M. Castelli, and L. Vanneschi, "Structural similarity index (SSIM) revisited: A data-driven approach," *Expert Systems with Applications*, vol. 189, p. 116 087, 2022. **DOI**: 10.1016/j.eswa.2021.116087.

Qingyu Liu was born in 1984 in Huainan, China. He received his MSc degree (2010) in computer software and theory from Anhui Normal University, China and the PhD degree (2023) in computer science from National University, Philippine. Between 2010 and 2016, he has worked as a computer software engineer in Huainan Coal Mining (Group) Co. Ltd. He is currently a Lecturer with the School of Computer Science, Huainan Normal University, Huainan, China. His research interests include Generative Adversarial Network, image inpainting and image processing.

Lei Chen received the MSc degree (2008) in computer science from Anhui University of Science & Technology, China. He is currently a Professor with the School of Computer Science, Huainan Normal University, Huainan, China. His research interests include knowledge graph, and deep learning.

Yeguo Sun received the MSc degree (2007) and PhD degree (2010) in control science and engineering from Beijing University of Aeronautics and Astronautics, China. He is currently a Professor with the School of Finance and Mathematics, Huainan Normal University, Huainan, China. His research interests include network control systems, neural networks and finite-time control. He has published more than fifty articles indexed by SCI and EI.

Lei Liu received his MSc degree (2013) in computer science from Anhui University of Science and Technology, China and the PhD degree (2023) in computer science from National University, Philippine. He is currently an Associate Professor with the School of Computer Science, Huainan Normal University, Huainan, China. His research interests include computer vision and machine learning.