Complex & Intelligent Systems
https://doi.org/10.1007/s40747-024-01411-5

ORIGINAL ARTICLE

# NLKFill: high-resolution image inpainting with a novel large kernel attention

Ting Wang[1] · Dong Xiang[1] · Chuan Yang[1] · Jiaying Liang[1] · Canghong Shi[2]

Received: 8 August 2023 / Accepted: 9 March 2024

## Abstract

The integration of convolutional neural network (CNN) and transformer enhances the network's capacity for concurrent modeling of texture details and global structures. However, training challenges with transformer limit their effectiveness to low-resolution images, leading to increased artifacts in slightly larger images. In this paper, we propose a single-stage network utilizing large kernel attention (LKA) to address high-resolution damaged images. LKA enables the capture of both global and local details, akin to transformer and CNN networks, resulting in high-quality inpainting. Our method excels in: (1) reducing parameters, improving inference speed, and enabling direct training on 1024×1024 resolution images; (2) utilizing LKA for enhanced extraction of global high-frequency and local details; (3) demonstrating excellent generalization on irregular mask models and common datasets such as Places2, Celeba-HQ, FFHQ, and the random irregular mask dataset Pconv from NVIDIA.

**Keywords** Image inpainting · Deep learning · Convolutional neural network · Transformer · Attention

## Introduction

Image inpainting focuses on reconstructing missing regions in a corrupted image using partially visible information, as illustrated in Fig. 1. Early approaches propagated information from neighboring visible regions [1–3], while recent research employs deep neural networks to directly generate plausible and visually coherent content and appearances [4–15].

In recent years, convolutional neural network (CNN)-based approaches [4, 8, 12, 16, 17] have dominated the field of image inpainting. By training on large-scale datasets, CNN can learn rich texture details and fill in missing regions with learned features. In addition, CNN is computationally efficient due to the sparse connectivity of convolutions. In addition, CNN has an inherent characteristic, namely translation invariance, which means that no matter how the object in the input image changes, the system should give the same output response, so that CNN can identify and extract features in images of any size. However, as the number of layers of the convolutional operation deepens, and the local inductive biases, these properties may no longer be beneficial and fail to adequately capture long-range interactions or global context [18]. This limitation hinders the ability of CNN to understand complex structures and relationships in a larger context.

Several related works [7, 9, 11–13] have presented various approaches to tackle the constraints imposed by CNN architectures. These approaches can be broadly categorized into two-stage methods, comprising a content inference module and an appearance refinement module. The two-stage methods generally first infer a coarse semantic image based on globally visible context, followed by supplementation with details in the second stage. However, the realization of

✉ Canghong Shi
canghongshi@163.com

Ting Wang
wangtingscu@126.com

Dong Xiang
fxdzcw@163.com

Chuan Yang
2962073857@qq.com

Jiaying Liang
1531483398@qq.com

[1] School of Computer Science, Chengdu University of Information Technology, Chengdu 610225, Sichuan Province, China

[2] School of Computer and Software Engineering, Xihua University, Chengdu 610039, Sichuan Province, China

Published online: 15 April 2024

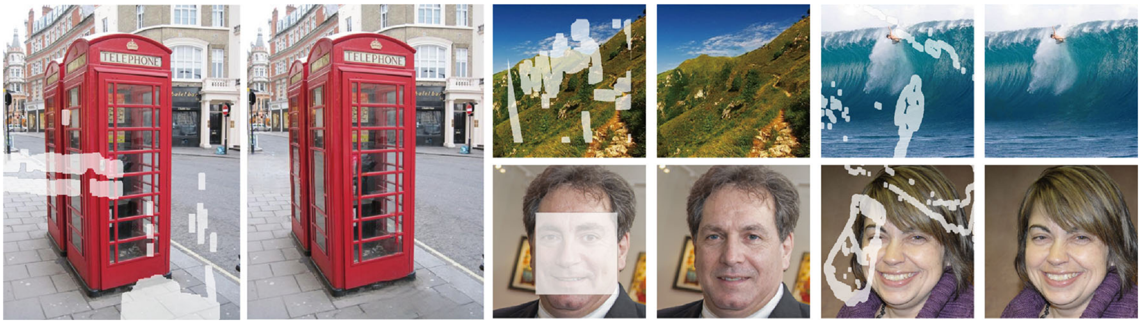


**Fig. 1** Selected image inpainting results of our proposed method under different scenes and different shapes of damaged areas (image damaged areas are displayed in white, and a certain percentage of transparency is set for better visualization)

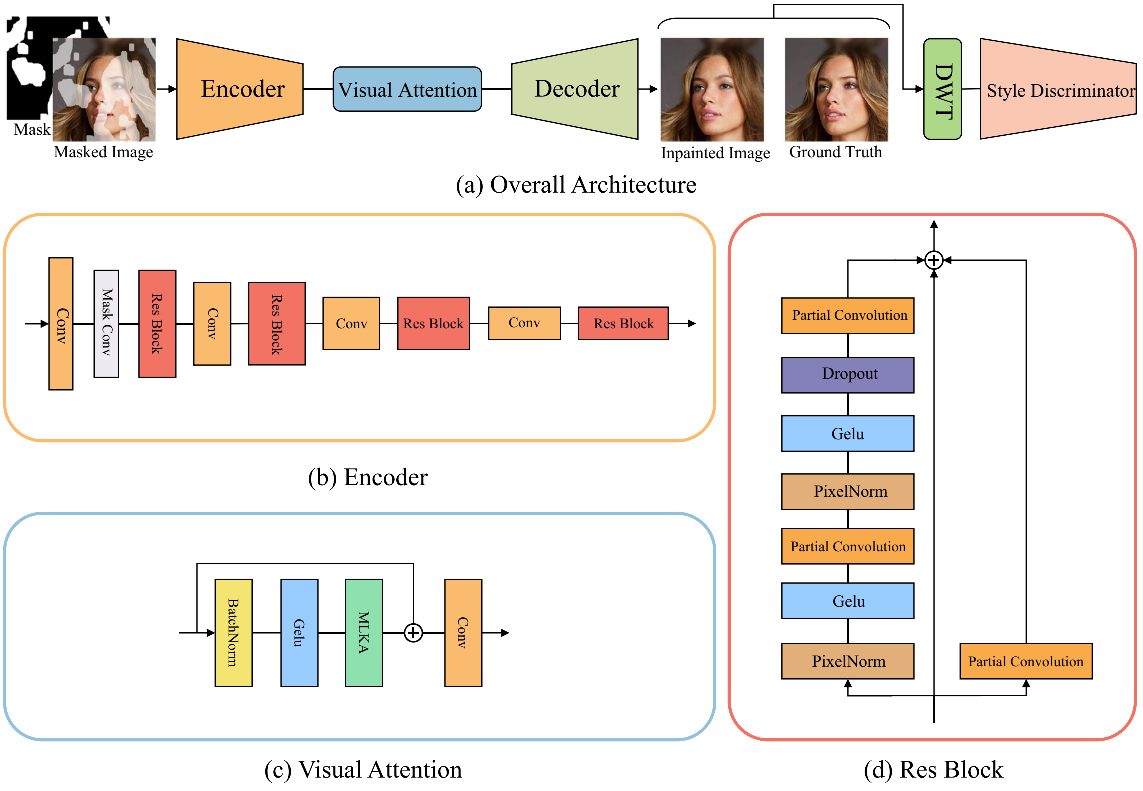
global contextual perception of these methods still fully relies on repeated local convolution operations, which amplifies the constraints imposed by the factor of translation invariance within CNN. The translation invariance of CNN often restricts information flow to mainly local regions, with global information being gradually shared through multi-layered heat propagation. As a result, the capability to capture global information is limited, which can lead to suboptimal performance in image processing tasks. In addition, when inferring content, the feature elements between adjacent network layers are interconnected through learned fixed weights, rather than input-adaptive weights, which may limit the deep features perception of the network. These issues make the transmission of long-distance features inefficient in very deep layers, resulting in the network tending to fill missing regions based on nearby rather than distant visible pixels.

Lately, the transformer architecture, known for its success in natural language processing (NLP), is increasingly being used in computer vision tasks. Unlike CNN models, the transformer model does not rely on local inductive priors and instead employs dense attention modules to capture long-range dependencies [19]. Some studies [20] have demonstrated the potential of the transformer in modeling structural relationships for natural image synthesis. Another advantage of the transformer is its ability to produce multiple outputs by optimizing the underlying data distribution. However, its quadratic increase in computational complexity with input length makes it challenging to use for high-resolution image synthesis or processing. In addition, most existing transformer-based generative models [20, 21] use an auto-regressive approach, which limits their application in image inpainting tasks where missing regions have arbitrary shapes and sizes.

Subsequently, research started on two-stage image inpainting methods based on the transformer. These methods typically combine CNNs and transformers to take advantage of the transformer's global structural understanding and the CNN's local texture refinement capabilities and computational efficiency. In the first stage, a transformer-based encoder is used to capture the structural features of the global context of the image to be inpainted. In the second stage, a CNN is employed to further supplement the image details. This two-stage inpainting approach has indeed alleviated some of the problems encountered in previous image inpainting tasks. Nevertheless, it still introduces considerable difficulty to the training process and presents some challenges.

With the goal in mind, an image inpainting network is developed, leveraging the large kernel attention (LKA) technique proposed by Guo et al. [22]. This methodology facilitates the efficient completion of high-resolution image inpainting tasks, such as those with dimensions of $512 \times 512$, all in a single stage. In NLP tasks, the transformer takes in one-dimensional sequences as inputs, whereas image data are three-dimensional and must be reshaped into a one-dimensional array to accommodate inputs of the transformer. For example, three-dimensional image data with a size of $256 \times 256 \times 3$ (where 3 represents the number of channels in an RGB color image) would result in a one-dimensional sequence of length 196,608. However, the largest model proposed in the previous study [19], namely the transformer-XL, has a fixed sequence length of 512, making training extremely challenging when exceeding this length. Fortunately, the LKA shares certain merits from CNN networks in terms of translation invariance and sliding window strategy, eliminating the necessity for considering image embedding representation or fixed sequences. Furthermore, the LKA possesses locality, long-distance dependence, and adaptability to spatial and channel dimensions, allowing it to construct both global and local feature representations, while avoiding the drawbacks of the transformer and the CNN network. Consequently, the LKA is better suited for high-resolution inpainting tasks. The paper presents several significant contributions as follows:

- The proposed method employs a single-stage image inpainting network that utilizes a LKA. This establishes a global correlation with the input features, effectively



(a) Overall Architecture

(b) Encoder

(c) Visual Attention

(d) Res Block

**Fig. 2** The overall architecture of our proposed method. **a** The method primarily consists of three parts: an encoder–decoder, a visual attention layer, and a progressive discriminator. **b** The encoder reduces the resolution of the input damaged image and continuously extracts shallow features through the dual-channel residual structure, as illustrated in (**d**) on the right. **c** This structure is the visual attention layer structure of the basic model NLKFill-B0. The feature content output from the encoder is transmitted to the visual attention layer, where the specific structure of modified large kernel attention module (MLKA) can be seen from Fig. 4(**a**). The attention distribution is calculated and subsequently fused with the input features to learn high-quality information from both the visible and invisible regions. This information is then parsed in the decoder network and finally outputs the inpainting result

reducing the limitations of the global semantic structure of the CNN network. Moreover, the original space and channel dimension structure of the image is retained, making it possible to complete the high-resolution image inpainting task.

- Experimental results indicate that the proposed single-stage image inpainting network based on the LKA outperforms existing state-of-the-art transformer-based image inpainting, whether two-stage or single-stage models, in terms of image inpainting quality, parameter quantity, inference speed, and memory usage.

## Related work

### Image inpainting

Traditional image inpainting methods mainly focus on filling in missing background pixels by copying and transferring visible pixels from other regions of the image. These methods include diffusion-based [1, 23, 24] and patch-based [2, 3, 25] techniques.

Propelled by the advancements in various generative adversarial networks (GANs) [26], conditional GANs (CGANs) [27], and variational autoencoders (VAEs) [28], a series of CNN-based approaches [4, 5, 7, 8, 12, 14] incorporating these techniques have emerged for semantic image inpainting. Adversarial learning was applied by Pathak et al. [8] for image inpainting of rectangular shaped holes. In addition, Iizuka et al. [4] extended this work [8] by introducing locally and globally consistent to handle random regular holes. Yu et al. [12] integrated the traditional patch-based idea with deep generative model-based networks. Partial convolution was proposed by Liu et al. [5] to deal with random irregular holes. Zheng et al. [14] explored multimodal image inpainting to produce diverse results. More and more research tend to incorporate auxiliary information of input images into models for a better image inpainting process. As an illustration, auxiliary edge information was incorporated by Nazeri et al. [7] into image inpainting. Furthermore, DeepFill v2

[29], Faceshop [30], SC-FEGAN [31] and MST [32] have combined more auxiliary information, such as contextual attention, gated convolution, and extracting lines and edges from the input images to project, for image inpainting. However, most above-mentioned models use similar CNN-based encoder–decoder networks, leading to the masked area being gradually influenced by adjacent visible pixels. Our model solves this problem by directly modeling the global context dependencies using LKA [22].

### Visual attention

Transformer was first proposed by Vaswani et al. [19] for machine translation, and later achieved success in various downstream NLP tasks. The overall network structure of transformer consists of stacked self-attention and point-wise feed-forward layers for both encoder and decoder. Due to its self-attention mechanism that can effectively capture the relevance between elements of the input sequence, transformer has become increasingly popular in the field of computer vision. For example, DETR [33] adopts transformer as the backbone to solve the problem of object detection. Dosovitskiy et al. [34] proposed ViT, which applied transformer to the field of image classification recognition for the first time, and achieved remarkable results compared with CNN-based methods. In addition, Parmar et al. [21] and Chen et al. [20] used transformer to complete image generation tasks. Nonetheless, the self-attention mechanism used in this method, originally designed for NLP tasks, encounters three key challenges when applied to computer vision tasks. To start with, it processes images as one-dimensional sequences, ignoring the inherent three-dimensional structure of images. Furthermore, it exhibits quadratic complexity, which proves to be computationally expensive for high-resolution images. Lastly, only spatial adaptability is realized, while the adaptability of channel dimension is ignored. In visual tasks, different channels usually represent different feature maps [35, 36]. Channel adaptability has been increasingly recognized as beneficial and crucial for visual tasks in recent years [36–40].

In contrast to these transformer methods that rely on a fixed position sequence length, which is unsuitable for repairing missing regions in irregular holes, this study introduces a novel visual attention method, i.e., the LKA. This method not only inherits the adaptability and long-distance dependence of the self-attention mechanism, but also capitalizes on the local context information inherent in traditional convolution. Our approach circumvents the need to address the issues of image embedding representation or fixed position sequence length, making it more appropriate for high-resolution image inpainting tasks.

## Methods

Image inpainting aims to restore an input image with missing pixels into an output image with complete pixels. Our goal is to employ multiple deep learning methods to jointly learn from an input high-resolution damaged image and finally obtain a high-resolution inpainting result.

To achieve a high-resolution repaired image, it is essential to employ a model that captures both global and local characteristics of the image. This capability ensures a reasonable coherence between local and global semantics. A novel single-stage LKA high-resolution imaging inpainting network called NLKFill is proposed. The default basic model used in our experiments is NLKFill-B0 and the overall architecture is shown in Fig. 2. In the later experiment, a larger model NLKFill-B1 is further designed to improve the model performance (for details please see the 4.4 Advanced experiment). First, a damaged image is encoded by a modified encoder to extract shallow features. Then, the output features from the encoder are passed as input features to the visual attention layer. The LKA used in this layer can capture the global attention of the image and also leverage the inductive bias of the CNN network to obtain the local features for detail filling. Subsequently, the high-resolution restored image that is nearly equivalent to the original can be obtained through the upsampling process of the decoder. Finally, the output result is fed into the discriminator, which evaluates it against a ground-truth sample to optimize the parameter weight within the image inpainting network.

### Visual attention layer

The attention mechanism can adaptively select and allocate relevant attention to features according to the input features. The key step of this mechanism involves generating attention maps that convey the importance of distinct segments. To accomplish this, it is essential to establish correlations among various features.

The current mainstream methods for generating attention maps involve either utilizing the self-attention mechanism in transformer [34, 41, 42] or large kernel CNN networks [38, 43–45] to capture long-distance dependencies and establish correlations among features. However, both of these methods have their own obvious drawbacks, especially when processing high-resolution images, which require a lot of computational resources and parameters.

Inspired by the visual attention network (VAN) proposed by Guo et al. [22], a visual attention layer is designed in this study to address the aforementioned issues, as illustrated in Fig. 2c. The key idea of the method is to capture long-distance dependencies by decomposing large kernel convolution operations. As shown in Fig. 3, large kernel convolutions can be divided into three convolutions, namely the depthwise local

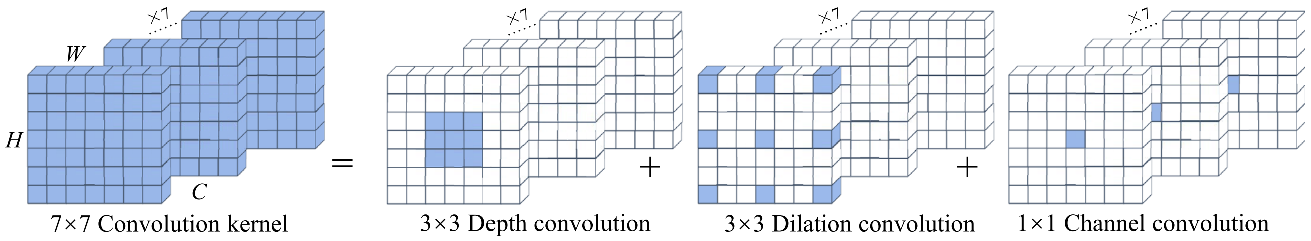


**Fig. 3** The standard large kernel convolution can be decomposed into three components: a depth-local convolution, a depth-dilated convolution, and a channel convolution. The figure shows how a 7 × 7 convolution is decomposed into a 3 × 3 depth-local convolution, a 3 × 3 depth-dilated convolution (with a dilation rate of 2), and a 1 × 1 channel convolution. $H$ and $W$ represent height and width, and $C$ denotes the number of channels. The dark grid signifies the position of the convolution kernel

convolutions, depthwise dilated convolutions, and 1 × 1 channel convolutions. More specifically, the decomposition of an LKA convolution with a 7 × 7 kernel size involves a depthwise local convolution with a 3 × 3 kernel size, a depthwise dilated convolution with a dilation rate $d$ of 2 and a 3 × 3 kernel size, along with a 1 × 1 channel convolution. The decomposition process is shown in Fig. 3, and the decomposition formulas are as follows:

$$d = \left\lfloor \frac{\lceil S/2 \rceil}{2} \right\rfloor \tag{1}$$

$$\begin{aligned} LKA_{S\times S} &= DepthConv_{(2d-1)\times(2d-1)} \\ &+ DilationConv_{\left\lceil \frac{S}{d} \right\rceil \times \left\lceil \frac{S}{d} \right\rceil} + Conv_{1\times 1} \end{aligned} \tag{2}$$

Through the utilization of the aforementioned decomposition, the model can capture long-distance global relations with reduced computational overhead and parameters. Upon obtaining long-distance global relations, the relative importance of each pixel can be estimated, and corresponding attention maps can be generated. The operation method of channel convolution $Conv_{1\times 1}$ is more similar to the fully connected layer, while the specific calculation method of depth-local convolution $DepthConv$ and depth dilation convolution $DilationConv$ for the input image is as follows:

$$DepthConv = \sum_{i=1}^{r}\sum_{j=1}^{r}\left(\sum_{m=1}^{2d-1}\sum_{n=1}^{2d-1} I_{ij}\left(m\cdot n\right)\cdot K\left(m,n\right)\right) \tag{3}$$

$$\begin{aligned} DilationConv &= \sum_{i=1}^{r}\sum_{j=1}^{r} \\ &\left(\sum_{m=1}^{\left\lceil \frac{S}{d} \right\rceil}\sum_{n=1}^{\left\lceil \frac{S}{d} \right\rceil} I\left(i+m\cdot d, j+n\cdot d\right)\cdot K\left(m,n\right)\right) \end{aligned} \tag{4}$$

where $S$ is the convolution kernel size, $r$ is the input image resolution size, $d$ is the dilated convolution rates, $K(m, n)$ represents the weight of the convolution kernel at position $(m, n)$, and $I$ represents the input image. The above convolution formula can calculate the feature map based on each pixel value of the input image.

Subsequently, we continued expanding upon the decomposition, further modifications are applied to $LKA$, incorporating a masked convolutional layer, $MaskConv$, designed to encourage the model to prioritize important visible values. In the $MaskConv$ procedure, the mask $m$ is initially assigned a floating-point value, and a kernel size of 2 × 2 is utilized. Focusing solely on extracting visible information accelerates the extraction of visible area features. Subsequently, the mask feature weight $X_m$ is obtained by flattening the mask $m_p$ as follows:

$$X_m = \begin{cases} \frac{\sum(m_p)}{S}, & \sum\left(m_p\right) > 0 \\ 0, & \sum\left(m_p\right) \le 0 \end{cases} \tag{5}$$

Therein, $m_p$ represents the mask pixel value, 1 represents the visible pixel, and 0 represents the masked pixel. The weight of masking $X_m$ is then applied to scale the attention score $A$. For example, if half of the image is masked, only the features of the remaining 50% of the visible area are extracted, and an initial mask weight of 0.5 is established. Finally, the updated original attention score $R$ is obtained through the decomposed $LKA$ procedure. The specific attention calculation formulas can be written as follows:

$$F_m = F \otimes X_m \tag{6}$$

$$D = DilationConv\left(DepthConv\left(F_m\right)\right) \tag{7}$$

$$A = Conv_{1\times 1}\left(D\right) \tag{8}$$

$$R = A \otimes F_m \tag{9}$$

where $F \in \mathbb{R}^{C\times H\times W}$ is the input feature, $F_m \in \mathbb{R}^{C\times H\times W}$ is the feature obtained after being processed by the masked convolutional layer $MaskConv$, $D \in \mathbb{R}^{C\times H\times W}$ represents the local structural information features, $A \in \mathbb{R}^{C\times H\times W}$ represents the attention map, and $R \in \mathbb{R}^{C\times H\times W}$ represents the final output fusion features. The values in the attention map rep-

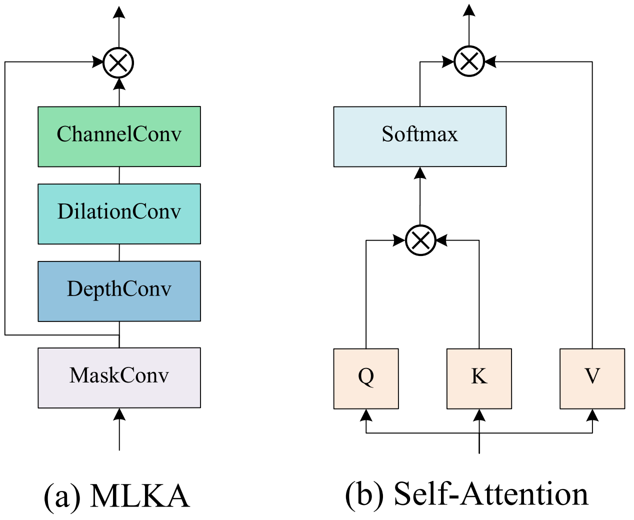


**Fig. 4** Model frame comparison diagram. **a** Modified large kernel attention module (MLKA). **b** Self-attention mechanism module in transformer

resent the importance of each feature, and ⊗ represents the element-wise product. It can be clearly seen that this method is different from common attention calculation methods. As shown in Fig. 4a, LKA [22] does not require the calculation of $Q$, $K$, $V$ and additional normalization like $softmax$ [19]. Besides obtaining the attention map through normalization, using the equivalent attention calculation method can also adaptively adjust the output according to the input features to obtain the attention distribution. Moreover, using the LKA not only combines the advantages of convolution and self-attention, but also realizes the adaptation of both the spatial dimension and the channel dimension. This further improves our fusion extraction of the global and local features of the input image.

## Encoder–decoder layer

A deep residual CNN [46] serves as the backbone of the encoder for extracting shallow image features, with its internal structure depicted in Fig. 2b. In the design of the residual structure, emphasis is placed on leveraging the strengths of the convolutional layer in initial image processing, facilitating the mapping of the image space to a higher dimensional feature space. A dual-channel parallel processing approach is adopted, as illustrated in Fig. 2d. In the first channel, each input image undergoes pixel normalization, followed by the application of the Gelu [47] activation function and two local convolutions. Simultaneously, the second channel directly performs a partial convolution on each input image. The final output results from the summation of the features from these two channels. In comparison to conventional CNN architecture, such a parallel structure allows for more stable optimization and significantly enhances the extraction of input image features.

In addition, the damaged regions of the image are separated in the encoder–decoder layer from the visible regions through an operation similar to masking the convolutional layer $MaskConv$. This ensures that each feature map representation exclusively captures locally visible information, thereby preventing cross-contamination of implicit correlations due to the large CNN sensory field. This approach not only enhances the sensitivity of the network compared to a normal CNN but also significantly improves computational efficiency.

Our decoder is basically the same as [10, 20, 48], and its structure is basically symmetrical to the encoder for upsampling to ensure stable image quality.

## Discriminator

To stabilize the training of the single-stage image inpainting network, a GAN framework is adopted, wherein the single-stage image inpainting network functions as the generator, and a discriminator architecture identical to StyleGAN [49] serves as the discriminator. The discriminator incorporates a progressive structure that more effectively guides the generator in producing high-resolution images. Inspired by StyleSwin [50] and SwaGAN [51], a discrete wavelet filter (DWT) [50, 51] is implemented before inputting images to the discriminator. This step is designed to capture high-frequency details and optimize the generator for high-resolution image inpainting, as illustrated in Fig. 2a and Fig. 5.

First, we apply pixel-level $L_1$ reconstruction loss [52] and perceptual loss [53] to the output of the inpainting network, forcing the generated image content to be closer to the basic truth and helping to judge whether the inpainted content is structurally reasonable. The reconstruction loss and perceptual loss are both improved on the basis of the $L_1$ loss method. Compared with the $L_1$ loss method, the reconstruction loss and the perceptual loss can better capture the high-level features of the image, and can be better for image generation tasks. It can better retain the semantic information of the image and better reflect the similarity between the original input and the generated output, but at the same time it may also introduce some additional computational costs. In addition, we introduce adversarial loss [27] to continuously improve the quality of repair. Through joint adversarial optimization of the generator and discriminator, we can achieve Nash equilibrium and ensure that the overall network model outputs high-quality results. The specific calculation formula of the above loss function is as follows:

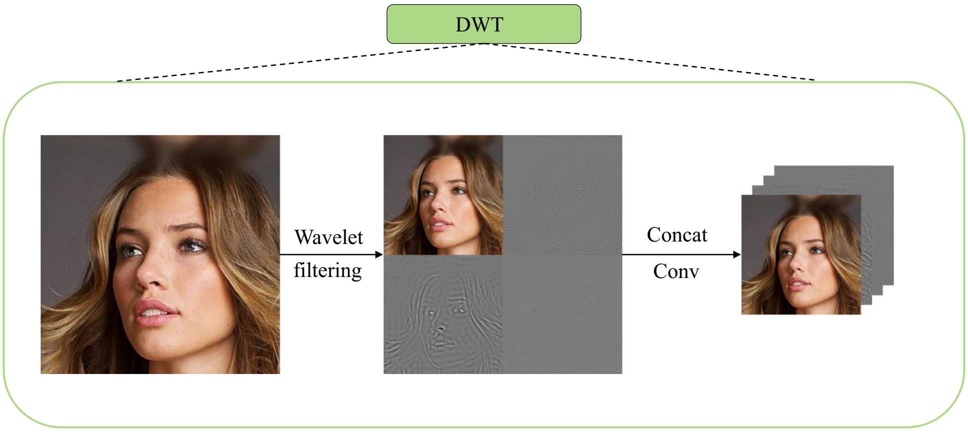


**Fig. 5** Capture directional multi-scale information of image texture through wavelet transform

$$L_{rec} = \left\| M_{real} - M_{fixed} \right\|_1 + \left\| I_{real} - I_{fixed} \right\|_1 + \left\| M \cdot \left( I_{real} - I_{fixed} \right) \right\|_1 \tag{10}$$

$$L_{per} = \frac{1}{N} \sum_{i=1}^{N} \left| \phi \cdot \left( I_{real} \right) - \phi \cdot \left( I_{fixed} \right) \right| \tag{11}$$

$$L_{GAN} = \log \left( 1 + exp \left( -D \left( I_{real} \right) \right) \right) \tag{12}$$

where $M_{fixed}$ and $M_{real}$ are the real image that represents the occluded area and the repaired image that represents the occluded area, respectively, $M$ represents the mask image, $I_{fixed}$ and $I_{real}$ are the original real image and the repaired image, respectively, $\phi$ is the visual geometry group (VGG) [54] pre-trained model, and $D$ is the discriminator.

Finally, the total loss function in this single-stage training can be expressed as Eq. (13):

$$L = L_{rec} + L_{per} + L_{GAN} \tag{13}$$

# Experiments

## Experimental details

### *Datasets*

We adopt three public datasets for imaging inpainting, namely the Celeba-HQ [55] and FFHQ [49] datasets for high-resolution face restoration, and Places2 [56] dataset for natural scenes. Brief introductions for each are provided as follows:

- Places2 comprises more than 1.8 million images captured across 365 diverse scenes. Owing to the intricate nature of these scenes, it stands out as one of the most demanding datasets for image restoration tasks. Our approach involves a standard train/test split, specifically 900 images per category are used for testing.
- Celeba-HQ stands as a high-quality dataset featuring human faces. The inclusion of high-frequency details in hairs and skin proves invaluable for assessing the fine-grained texture synthesis capabilities of models. Our training dataset comprises 29,000 images, with an additional 1000 images earmarked for testing.
- FFHQ is also a high-quality dataset featuring human faces. This dataset contains a large number of face images from different races, ages, and genders. It has broad diversity, making it suitable for various face-related research. Our training dataset contains 68,000 training images and 2000 designated test images.

Since a large number of real damaged images cannot be collected, extra masks are added to the original data to obtain occluded damaged images. The Pconv [5] mask dataset released by NVIDIA, which includes rectangular and irregular masks, is chosen here to enhance the effectiveness of network model training. In the present experiment, we adopted 55,000 irregular masks in the training set, and 12,000 irregular faces in the test set.

### *Baselines*

We have listed most of the current excellent and famous models for comparison with their abbreviations and brief introductions as follows:

- PIC [14] adopts a multivariate image completion approach to the task of generating multiple different reasonable solutions for image completion.
- DeepFillv2 [29] designs gated convolutions to solve the problem of ordinary convolutions, which can use free-form masks and guides to complete the image.
- HiFill [11] proposes a lightweight model with an efficient contextual residual aggregation mechanism to achieve ultra-super-resolution image restoration.
- CRFill [13] teaches this patch borrowing behavior to attention-free generators through joint training on the

auxiliary context reconstruction task, which makes the generated output reasonable even when reconstructed from surrounding regions, as well as capturing the correspondence between missing and known regions.

- ICT [10] is a coarse-to-fine two-stage inpainting network that uses a combination of transformer and convolutional neural network to complete multivariate image.
- TFill [15] is also a two-stage inpainting network from coarse to fine. It uses a restricted convolutional neural network with small and non-overlapping receptive fields for weighted label representation, which allows the transformer to explicitly model remote visible contextual relationships of equal importance in all layers, while using larger receptive fields do not implicitly confuse adjacent markers.

### *Metrics*

In this study, we primarily use traditional patch-level image quality metrics, including peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM), as well as the latest image similarity metrics, such as Learned Perceptual Image Patch Similarity (LPIPS) [57] and Fréchet Inception Distance (FID) [58].

PSNR is calculated by the mean square error between the original image $I_{real}$ and the processed image $I_{fixed}$, and can be used to objectively evaluate image quality and image similarity. The formula is as follows:

$$PSNR\left(I_{real}, I_{fixed}\right) = 20\left(\frac{2^n-1}{\sqrt{MSE\left(I_{real}, I_{fixed}\right)}}\right) \tag{14}$$

where $n$ is the number of bits representing a pixel. The mean squared error ($MSE$) is computed as the expected value of the square of the difference between the estimated parameter value and the true parameter value, and it can be expressed using Equation (15). The $r$ in $MSE$ represents the image input resolution:

$$MSE = \frac{1}{r^2}\sum_{i=1}^{r}\sum_{j=1}^{r}\left(I_{real}\left(i, j\right) - I_{fixed}\left(i, j\right)\right)^2 \tag{15}$$

SSIM is another index to measure the similarity of two images. When one of the two images is an undistorted image and the other is a distorted image, the SSIM value of the two images can be regarded as a quality indicator of the distorted image. The larger the value of SSIM, the higher the similarity between the two signals. Compared with PSNR, SSIM is more in line with the judgment of human eyes on image quality in the measurement of image quality. The formula is as follows:

$$SSIM\left(I_{real}, I_{fixed}\right) = \frac{\left(2\eta_{I_{real}}\cdot\eta_{I_{fixed}}+c_1\right)\left(2\sigma_{I_{real}I_{fixed}}+c_2\right)}{\left(\eta_{I_{real}}^2+\eta_{I_{fixed}}^2\right)\left(\sigma_{I_{real}}^2+\sigma_{I_{fixed}}^2+c_2\right)} \tag{16}$$

where $\eta_{I_{real}}$ and $\eta_{I_{fixed}}$ are the brightness mean values of the input image $I_{real}$ and $I_{fixed}$ respectively, $\sigma_{I_{real}}$ and $\sigma_{I_{fixed}}$ are the brightness variance of the input image $I_{real}$ and $I_{fixed}$ respectively, $c_1$ and $c_2$ are two constants used to stabilize the calculation. They are used to prevent division by zero in the denominator and to prevent instability caused by too small values in the denominator, respectively.

The FID serves as a metric to depict the diversity and quality of generated images. It entails the calculation of feature space statistics for both the real and generated images. A reduced FID value indicates improved image diversity and quality. The formula is as follows:

$$FID = \left\|\mu_{I_{real}} - \mu_{I_{fixed}}\right\|^2 + T_{I_{real}}\left(\sum I_{real} + \sum I_{fixed} - 2\sqrt{\sum I_{real}\cdot\sum I_{fixed}}\right) \tag{17}$$

where $\mu_{I_{real}}$ and $\mu_{I_{fixed}}$ are the real images $I_{real}$ and generate the mean vector of the image $I_{fixed}$ in the feature space respectively, $\left\|\mu_{I_{real}} - \mu_{I_{fixed}}\right\|^2$ represents the Euclidean distance between mean vectors squared, $T_{I_{real}}$ is the trace of the covariance matrix in the feature space of the real image, $\sum I_{real}$ and $\sum I_{fixed}$ denote the traces of the covariance matrices of the real image and the generated image in the feature space, respectively, and $\sqrt{\sum I_{real}\cdot\sum I_{fixed}}$ represents the square root of the feature space covariance matrix of the real image and the generated image.

Compared to the FID, which is generally used for deeping features measure image similarity, LPIPS is more in line with human perception than traditional methods, and the lower the value of LPIPS, the more similar the two images are. The basic idea of LPIPS is to use the features of deep networks to measure the perceptual distance between images instead of pixel distance. The features of deep networks can capture high-level semantic information of images instead of low-level detailed information. Therefore, LPIPS can better reflect human visual perception than simple mean square error (MSE) or peak signal-to-noise ratio (PSNR). The formula is as follows:

$$LPIPS\left(I_{real}, I_{fixed}\right) = \sum_{i=1}^{L}\frac{1}{H_iW_iC_i}\sum_{h,w,c}\left(\phi_i\left(I_{real}\right)_{h,w,c} - \phi_i\left(I_{fixed}\right)_{h,w,c}\right)^2 \tag{18}$$

**Table 1** Quantitative comparison on the Places2 and Celeba-HQ test sets using different ratios of irregular masks with an input image resolution of 256 × 256. NLKFill-B0 can achieve a similar level of inpainting as TFill with a lower cost. This method surpasses the latest method TFill because it only trains in a single stage, which is simpler and easier than the two-stage approach. In these metrics, ↓ indicates lower is the better, and ↑ indicates higher is the better

| Dataset | Mask Ratio | LPIPS↓ | | FID↓ | | PSNR↑ | | SSIM↑ | |
|---|---|---|---|---|---|---|---|---|---|
| | | 20–40% | 40–60% | 20–40% | 40–60% | 20–40% | 40–60% | 20–40% | 40–60% |
| Places2 | PIC | 0.1425 | 0.1956 | 25.60 | 35.59 | 23.38 | 21.51 | 0.8185 | 0.7484 |
| | DeepFillv2 | 0.1437 | 0.1859 | 26.03 | 32.04 | 22.54 | 20.72 | 0.8015 | 0.7393 |
| | HiFill | 0.1945 | 0.2656 | 32.65 | 47.32 | 21.34 | 19.31 | 0.7447 | 0.6625 |
| | CRFill | 0.1393 | 0.1795 | 20.32 | 26.51 | 23.16 | 21.19 | 0.8229 | 0.7612 |
| | ICT | 0.1304 | 0.1769 | 19.76 | 25.29 | 23.68 | 21.97 | 0.8297 | 0.7611 |
| | TFill | 0.1128 | 0.1562 | 17.64 | **22.53** | **23.99** | 21.87 | 0.8374 | 0.7727 |
| | NLKFill-B0 | **0.1123** | **0.1530** | **17.28** | 23.41 | 23.17 | **22.01** | **0.8380** | **0.7801** |
| Celeba-HQ | PIC | 0.0845 | 0.0904 | 14.513 | 25.031 | 26.781 | 21.723 | 0.9330 | 0.8112 |
| | DeepFillv2 | 0.0680 | 0.0702 | 16.278 | 28.711 | 25.868 | 21.108 | 0.9221 | 0.8020 |
| | ICT | 0.0701 | 0.0766 | 10.515 | 20.843 | 28.242 | 23.076 | 0.9522 | 0.8641 |
| | TFill | 0.0562 | 0.0622 | **10.386** | 20.167 | 32.761 | 31.458 | 0.9672 | 0.9518 |
| | NLKFill-B0 | **0.0557** | **0.0615** | 10.470 | **20.066** | **33.088** | **31.773** | **0.9691** | **0.9573** |

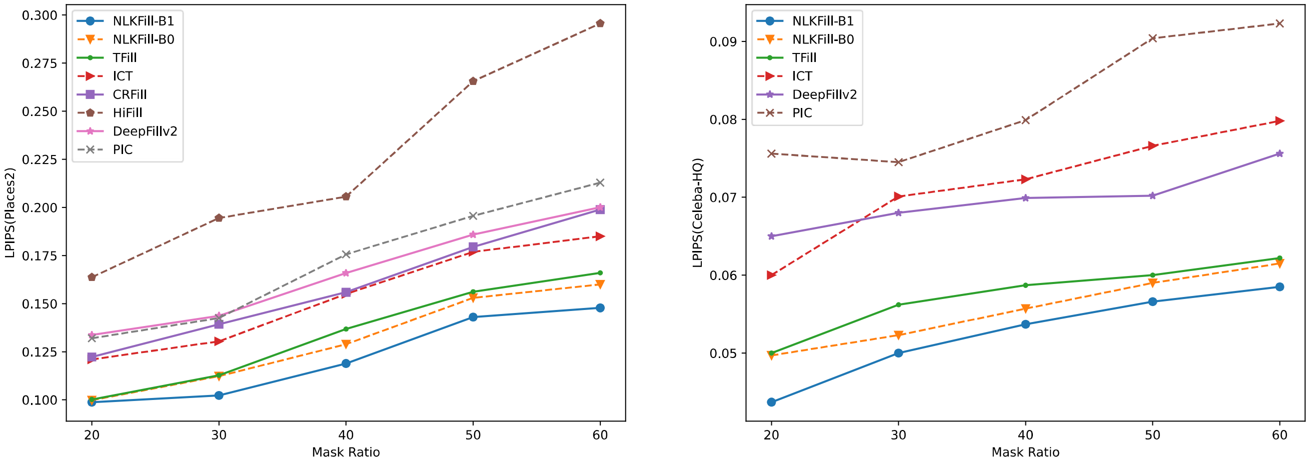


**Fig. 6** The quality comparison of the repair results of various methods under different mask ratios can be observed. Among them, the evaluation of the LPIPS metric is primarily carried out using the Places2 and Celeba-HQ datasets. The quality improves as the metric value decreases

where $I_{real}$ and $I_{fixed}$ are two images, $L$ is the number of layers of the deep network, $\phi_i$ is the feature extraction function of the i-th layer, and $H_i, W_i, and C_i$ are the height, width and number of channels of the i-th layer, respectively.

### *Implementation details*

The model uses PyTorch v1.9.1 and three NVIDIA A4000 GPUs (16 GB) on the Ubuntu 18.0 LTS system. Based on the experimental requirements, the single-stage image inpainting network is trained on images with resolutions of 256 × 256, 512 × 512, 1024 × 1024, and the corresponding batch sizes during training are 48, 12 and 4. The time for initial step training is set as 5,00,000 iterations, and we adopt an Adam optimizer with beta1 = 0.5 and beta2 = 0.9. For the learning rate schedule, we set its initial value as 0.0002 for the first 2,00,000 iterations and linearly decay it to zero in the next 3,00,000 iterations. The loss optimization function of this network is $L = L_{rec} + L_{per} + L_{GAN}$.

## Main results

We first compare our basic model NLKFill-B0 with the following state-of-the-art image inpainting methods, namely PIC [14], DeepFillv2 [29], HiFill [11], CRFill [13], ICT [10] and the latest TFill [15]. We use their publicly released code and models for the experiments.

### *Quantitative comparison*

Table 1 shows the quantitative evaluation results on Places2 [56] and Celeba-HQ [55] test sets, using the irregular masks

**Table 2** TFill requires a two-stage process from coarse to fine. The NLKFill-B0 can obtain inpainting results with a single-stage training

| Model | Input size | Batch size | FLOPs (G) | Param (M) | Mem (M) | GPU infer time (ms) |
|---|---|---|---|---|---|---|
| TFill-coarse | 256 | 12 | 10.23 | 83.229 | 13,467 | 49.70 |
| NLKFill-B0 | 256 | 12 | **1.15** | **47.804** | **10,347** | **28.97** |
| TFill-refine | 512 | 3 | 16.01 | 147.553 | 13,635 | 57.88 |
| NLKFill-B0 | 512 | 3 | **1.15** | **47.804** | **11,099** | **39.80** |

provided in the test set of Pconv [5]. The mask ratio represents the range of mask ratios applied to the image. The original mask ratio is divided into 20%–40% and 40%–60%. In accordance with the experiments of ICT [10] and TFill [15], we only compare the results on the mid-level mask ratio (mainly concentrated in 30% and 50%). In Fig. 6, the quality comparison of repair results for various methods under different mask ratios is observable, with the primary evaluation conducted using the LPIPS metric. The results show that our model surpasses the previous state-of-the-art CNN-based models at all mask scales. Our basic model achieves a similar performance as TFill, but TFill needs to go through two stages from coarse to fine. In contrast, we use a simpler structure, a lower training cost, and only one stage to complete the inpainting task, as shown in Table 1. Figure 10 also clearly demonstrates that our method outperforms TFill in terms of inpainting details.

Although our basic model NLKFill-B0 metrics does not perform as well as TFill in under certain conditions, such as the FID (40%–60%) and PSNR (20%–40%) indicators of Places2, and the FID (20%–40%) indicator in the Celeba-HQ dataset in Table 1. This may be related to the technology used in the measurement of each index. For example, FID is a measure used to calculate the distance between the real image and the feature vector of the generated image. The smaller the FID value, the higher the similarity. From the experimental results, NLKFill is more inclined to produce natural and reasonable restorations. Therefore, the structure of the real image may sometimes not be fully followed in some areas. More details of the comparison are provided in Appendix Fig. 15.

However, it is worth noting that both ICT and TFill in recent years use transformer to capture global information through a two-stage method, in which ICT directly downsamples the original image to 32 × 32 or 48 × 48 resolution, resulting in important information lost in this straightforward massive downsampling process. NLKFill-B0 only trains in a single state, which is more simple and easier than the two-stage approaches. In addition, our further experiments (for details please see the 4.4 Advanced experiment) demonstrates that when we modify the model to a larger size model (NLKFill-B1), which has a slightly smaller model size than TFill, the image repair effect of our model is better than that of TFill model on the whole Table 2 .

***Qualitative comparison***

To make the comparison of inpainting results of various image inpainting methods more intuitive, we use the same input data for qualitative evaluation. Figure 7 shows images of natural scenes occluded by random masks. In this case, we mainly compare the results of semantic content inpainting. Among them, PIC [14], DeepFillv2 [29] and HiFill [11], although good at removing the target object, fail to infer the correct shape after inpainting the object. For example, in the road images (please see the second and third rows in the Fig. 7), they cannot correctly infer the road behind the object. In the animal images, only DeepFillv2, CRFill [13], TFill [15] and our method can infer the dog's legs. In addition, our method generates more realistic results in terms of the dog's head appearance and background content reasoning, which are more reasonable compared to other methods. In the removal of grass shadows, our method also provides more reasonable inpainting results. The most challenging task is to inpaint the damaged area of the structured house. It can be observed that the inpainting of the random rectangular damaged area in Fig. 7 is not satisfactory for any method. However, even so, our method still performs better than the previous methods, producing semantic structures closer to real images in damaged regions, rather than producing too many unnaturally distorted structures.

The previous inpainting methods have tolerably slight flaws in inpainting damaged face images at a resolution of 256 × 256, as shown in Fig. 8. However, when the image resolution is increased to 512 × 512, these flaws become more noticeable, and even make these inpainting methods fail to complete the inpainting task, as shown in Fig. 9.

The latest method TFill uses a two-stage design to successfully inpaint damaged face images with a resolution of 512 × 512. However, our method can produce more natural and reasonable inpainting results than TFill when important semantic information is missing. As shown in Fig. 10, our method can handle the transition of the hair region more naturally and reasonably, and does not generate vertical streaks caused by stitching the inpainting results onto the damaged images. More comparisons are presented in Appendix 15.

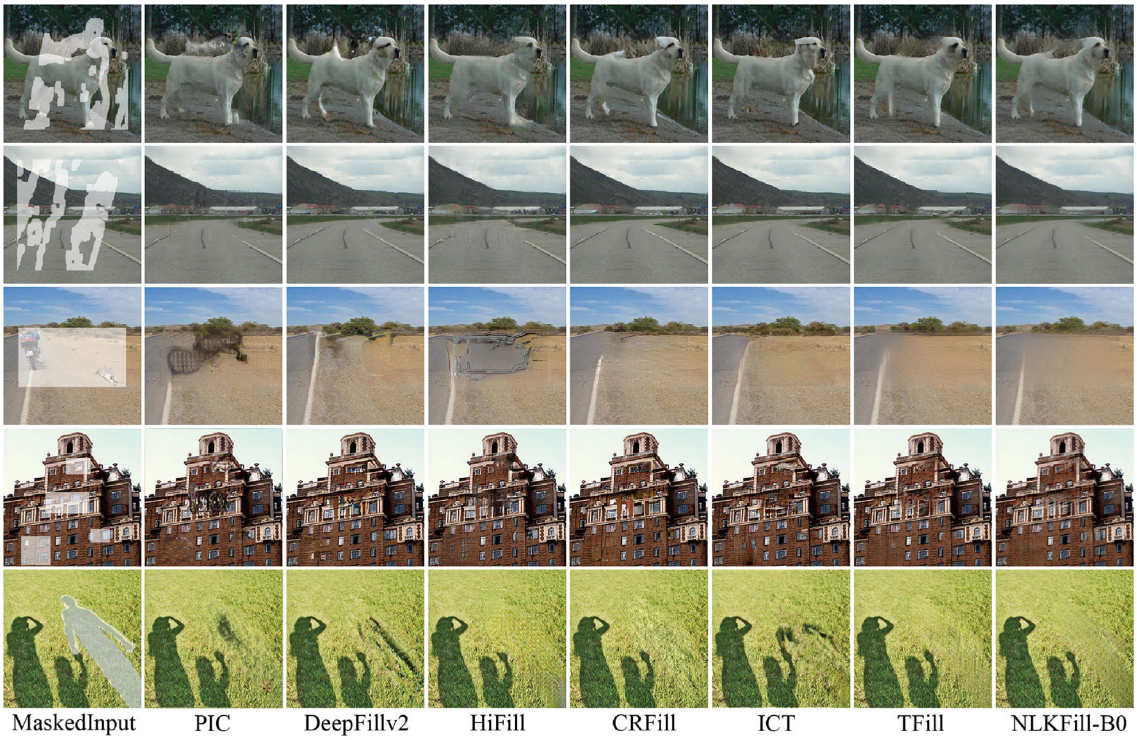


**Fig. 7** Qualitative comparison and inpainting results of Places2 test set with random irregular mask and rectangular mask, the image resolution is used 256 × 256, because earlier work can only be trained at this scale. Our model generates more reasonable object and scene structures with better visual effects than previous methods

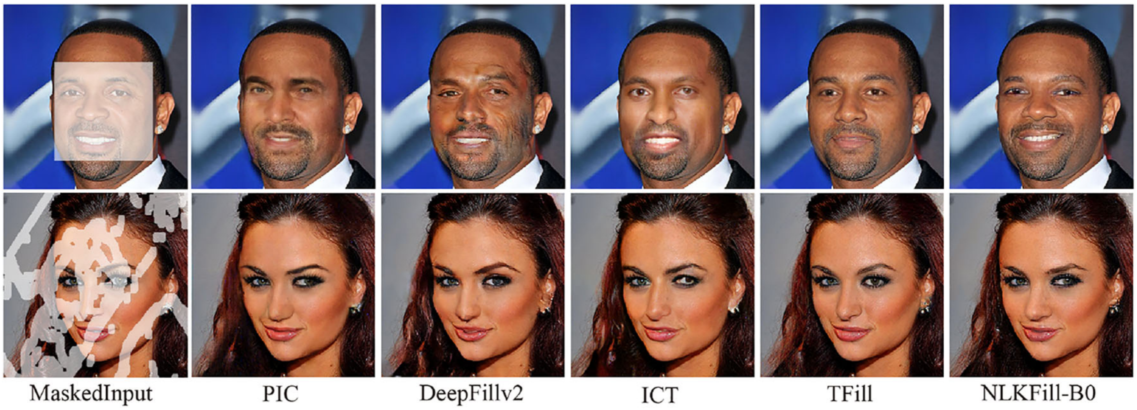


**Fig. 8** On the image resolution scale of 256×256, whether it is a random irregular mask or a rectangular mask, there is only a slight gap in the performance of each method on the Celeba-HQ test set. Our model has nearly the same repair quality as TFill

## Ablation experiments

Ablation experiments on our model are conducted to assess the impact of each module of our method on the image inpainting results. The results with and without the dual-channel residual layer, modified encoder–decoder layer, progressive discriminator and LKA layer are compared. In addition, the effect of the wavelet filter [50, 51] on the image inpainting quality is also analyzed. The results are shown in Table 3.

We carry out this ablation experiment on the Celeba-HQ [55] and Places2 [56] datasets, respectively, to train and evaluate our model. We start with the encoder–decoder structure of VQGAN [48] as a baseline model, and then introduce a dual-channel residual module. The experiment demonstrates that this structure design significantly improves the performance. When we further integrate this model with

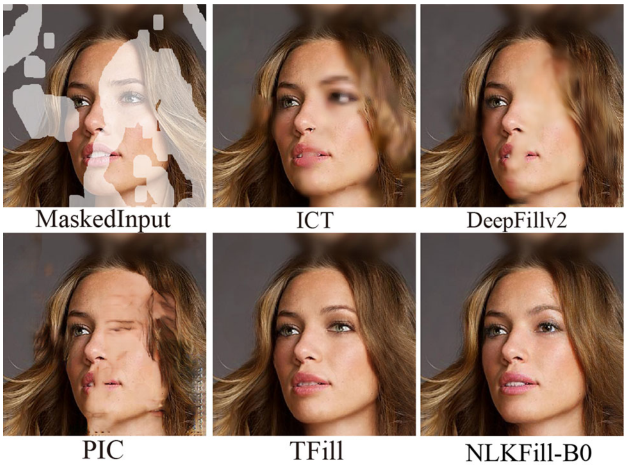


**Fig. 9** Comparison of methods using the 512 × 512 resolution Celeba-HQ testset

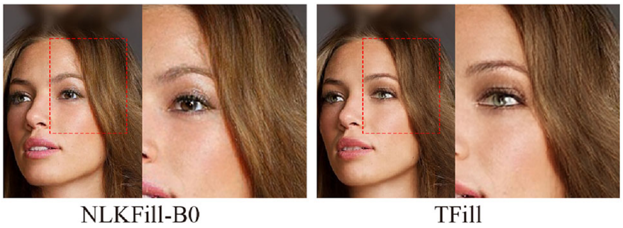


**Fig. 10** Comparison with TFill method in terms of inpainting details. For example, TFill produces vertical stripes on the hair due to stitching, while our method makes the transition more natural

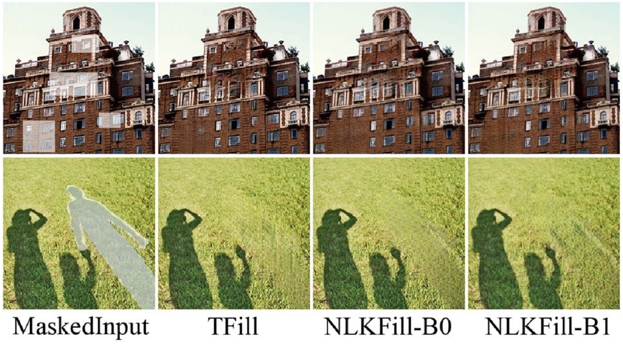


**Fig. 11** A continuation of Fig. 7, using the Places2 test set with a resolution of 256×256 as input, and showing some more results. It can be seen that the inpainting results of NLKFill-B1 are slightly better than NLKFill-B0, but it also shows that NLKFill still needs to further improve on the Places2 dataset to enhance semantic understanding

the discriminator of StyleGAN [49], its performance already becomes comparable to that of previous CNN-based methods.

In addition, we further optimize the encoder–decoder layer by employing a modified convolutional network layer to effectively separate the damaged and visible regions in the image. The results in Table 3 show that our improvement leads to a significant performance boost, especially for the FID [58] metric. This demonstrates the importance of applying different weights for visible and damaged regions, as opposed to using general convolutional networks in image inpainting tasks. On top of that, we add wavelet filters to further enhance the quality of the generated results.

**Table 3** The ablation studies for each module, which finally validates the effectiveness of the modules. In these metrics, ↓ indicates lower is the better, and ↑ indicates higher is the better

| Method | Celeba-HQ | | Places2 | |
|---|---|---|---|---|
| | LPIPS↓ | FID↓ | LPIPS↓ | FID↓ |
| Baseline | 0.0587 | 11.17 | 0.1181 | 18.15 |
| +Style Discriminator | 0.0585 | 11.10 | 0.1175 | 18.14 |
| +Deep Two-Channel Residual | 0.0573 | 11.03 | 0.1143 | 17.88 |
| +Modified Encoder–Decoder | 0.0559 | 10.87 | 0.1138 | 17.53 |
| +DWT | 0.0554 | 10.71 | 0.1121 | 17.48 |
| +MLKA | **0.0542** | **10.44** | **0.1104** | **16.97** |

**Table 4** NLKFill-B0 is the basic model, and NLKFill-B1 is a version similar to the model size of the TFill method. Layers indicates the number of visual attention layers and the MLKA modules contained in them. The FLOPs data are obtained from testing under the RTX A4000 GPU hardware environment using 256 × 256 resolution image input

| Model | Layers | Param (M) | FLOPs (G) |
|---|---|---|---|
| NLKFill-B0 | {1, 0, 0, 0} | 47.804 | 1.15 |
| NLKFill-B1 | {3, 4, 5, 3} | 89.113 | 8.68 |

Finally, we adopt an enhanced LKA, which further boosts the inpainting quality and metric scores. The employment of this attention mechanism highlights its significant role in improving the performance of our model.

The above-mentioned experiments demonstrate that the addition and application of each module constitute an effective improvement to the model for this image inpainting task. This is especially true when the image dataset contains high-frequency content or fine details.

## Advanced experiment

Due to the simple hierarchical structure of NLKFill, we extend the NLKFill-B1 model based on NLKFill-B0 to further address the performance challenges of tasks with different complexity levels. In comparison, NLKFill-B1 has a four-layer structure with decreasing resolution, namely $\frac{H}{4} \times \frac{W}{4}, \frac{H}{8} \times \frac{W}{8}, \frac{H}{16} \times \frac{W}{16}, \frac{H}{32} \times \frac{W}{32}$, and twice the number of parameters. Here, H and W denote the height and width of the input image respectively. As the resolution decreases, the number of output channels C will continue to increase. The detailed information of each model is shown in Table 4 and the model structure of NLKFill-B1 is shown in Fig. 12.

We present the experimental training results for NLKFill-B1 on the Places2 [5] and FFHQ [49] test sets in Table 5, and

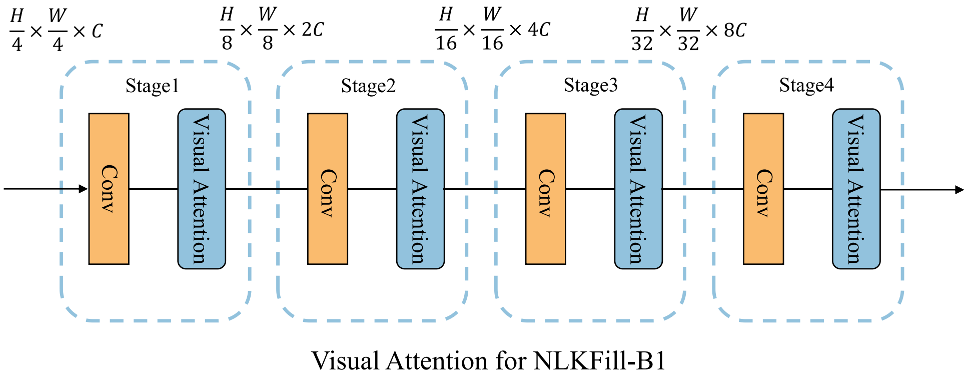

$\frac{H}{4} \times \frac{W}{4} \times C$
$\frac{H}{8} \times \frac{W}{8} \times 2C$
$\frac{H}{16} \times \frac{W}{16} \times 4C$
$\frac{H}{32} \times \frac{W}{32} \times 8C$
Stage1
Stage2
Stage3
Stage4
Conv
Visual Attention
Visual Attention for NLKFill-B1

**Fig. 12** This structure is the middle layer (four-stage visual attention layer) structure of the larger model NLKFill-B1. The feature content output by the encoder goes through four stages of visual attention layers, and the resolution of each layer decreases step by step. Except for the middle layer of the network, the rest of the structure is consistent with NLKFill-B0, see Fig. 2 for details. The specific structure of the visual attention layer in each stage can be seen in Fig. 2(c)

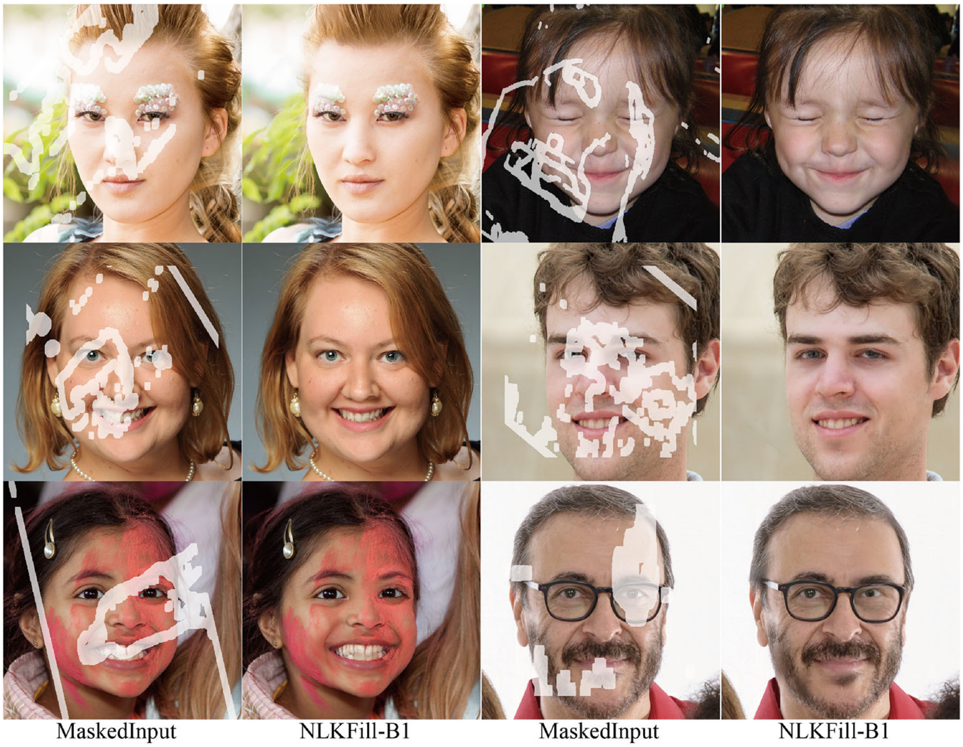


**Fig. 13** Our model can train directly on images with a resolution of 1024 × 1024 and output inpainting results with the same resolution. The results show that NLKFill-B1, which has a larger model size, can capture and utilize more detailed features in high-resolution images. Due to limitations in computing resources and training time, the output is not optimal yet, but it demonstrates the future potential of our method for high-resolution inpainting tasks

the mask ratio setting is a modest of 40%. We pay particular attention to the performance of inpainting results on the Places2 dataset. The table shows that the larger the model, the better the indicators and the Fig. 11 illustrates that enlarging the model improves the inpainting results of the Places2 test set to some extent. Additional inpainting results at different resolutions are provided in Appendices 16 and 17.

Furthermore, the model's simple hierarchical structure allows it to be trained directly with uncompressed 1024 × 1024 high-resolution images as input. To better control over the resolution of input images, we only use the model NLKFill-B1 and the face dataset FFHQ for training (FFHQ dataset contains 70,000 high-definition face images of 1024× 1024 resolution). Figure 13 shows that NLKFill-B1 still captures enough detailed features from high-resolution images, making the restoration structure more refined and natural, and finally outputs high-resolution inpainting results. The quality comparison of the repair results of various methods under different output resolutions can be observed in Fig. 14 and some other high-resolution inpainting results are provided in Appendix 17.

The advanced experimental results confirm that larger models indeed achieve superior performance. However, larger models also require more training time, larger and higher quality datasets, especially for high-resolution inpainting tasks like 1024 × 1024 resolution. Due to computational

**Table 5** Comparing the effects of different size models at different resolutions. In these metrics, ↓ indicates lower is the better, and ↑ indicates higher is the better

| Model | Input size | FFHQ | | Places2 | |
|---|---|---|---|---|---|
| | | LPIPS ↓ | FID ↓ | LPIPS ↓ | FID ↓ |
| TFill | 256 | 0.0591 | 15.23 | 0.1328 | 20.12 |
| NLKFill-B0 | 256 | 0.0586 | 15.27 | 0.1329 | 20.34 |
| NLKFill-B1 | 256 | **0.0571** | **15.15** | **0.1323** | **19.94** |
| TFill | 512 | 0.0621 | 15.11 | 0.1339 | 20.31 |
| NLKFill-B0 | 512 | 0.0592 | 15.58 | 0.1342 | 20.75 |
| NLKFill-B1 | 512 | **0.0565** | **14.84** | **0.1310** | **19.74** |
| NLKFill-B1 | 1024 | 0.0549 | 15.53 | – | - - |

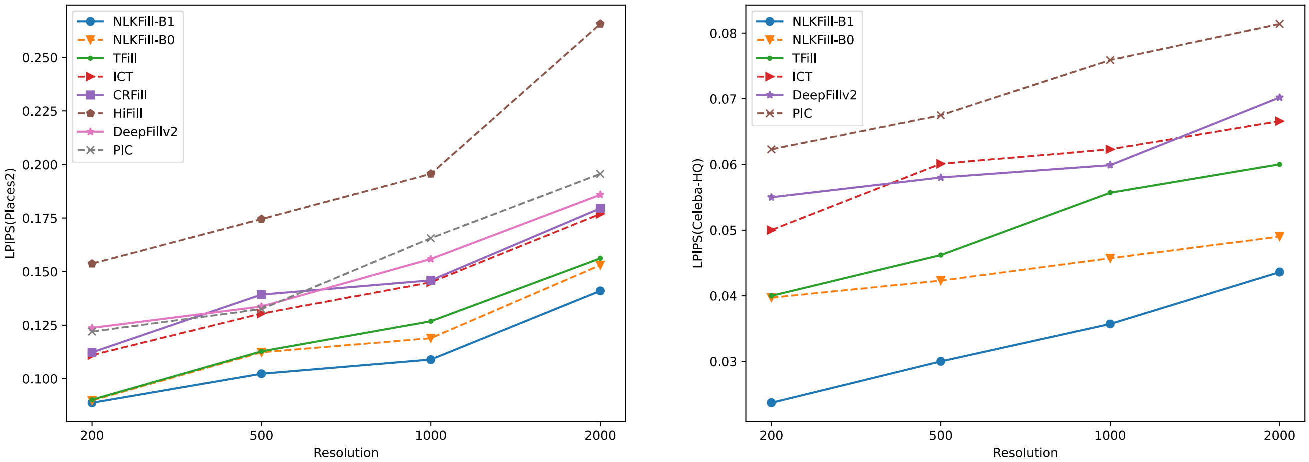


**Fig. 14** The quality comparison of the repair results of various methods under different output resolutions can be observed in the figure. Among them, the evaluation of LPIPS metric is mainly carried out using the Places2 and Celeba-HQ datasets. Quality increases as the metric decreases

resource constraints, we have not yet fully explored the model's potential and showcased its best results. Nevertheless, based on its current performance, the model still offers a lot of room for improvement.

## Conclusion and future work

In this study, we introduced the large kernel attention (LKA) mechanism as a replacement for traditional network architectures, combining the global structural modeling capabilities reminiscent of transformers with the local detail feature capture abilities of CNNs. It is crucial to emphasize that the LKA mechanism is not a transformer network; rather, it operates based on large kernel convolutions.

Through innovative enhancements to the encoder–decoder layer and the LKA layer, we dynamically balanced the weights of visible and damaged regions, resulting in an improved quality of inpainted images.

Experimental results demonstrate that our proposed method outperforms previous single-stage or two-stage transformer-based inpainting networks across various metrics, including parameter quantity, inference speed, memory usage, and computing resource overhead. This advancement facilitates the direct training of high-resolution images, such as 512 × 512 and 1024 × 1024. Furthermore, we conducted ablation experiments on different modules of NKFill, exploring and evaluating the model's potential and areas for improvement.

Despite these achievements, there is room for improvement, particularly on the Places2 dataset. The inpainting performance on irregularly damaged images of different scene types and complex scenes is slightly inferior to that on facial images. This indicates the necessity for further exploration of comprehensive understanding and utilization of semantic content in images. Our ongoing work and future research will focus on refining the model's performance across various datasets, enhancing semantic content understanding, and pushing the boundaries of high-resolution image inpainting.

**Acknowledgements** Thanks for the key research and development projects of Sichuan Science and Technology Department project (2023YFG0099, 2023YFG0261).

**Author Contributions** Ting Wang: experimental design, image data analysis, results interpretation, manuscript drafting, manuscript revi-

sion; Dong Xiang: image data analysis, manuscript drafting, results interpretation, Grammatical error correction; Chuan Yang and Jiaying Liang: data acquisition and image data analysis; Canghong Shi: conception and design of the study, results interpretation, manuscript revision; all the authors read and approved the final manuscript.

**Data availability** The face datasets and natural scene datasets used in this article were obtained from open-source links. Researchers in this field have integrated them, and the FFHQ dataset is accessible at https://github.com/NVlabs/ffhq-dataset, while the Celeba-HQ dataset is available at https://github.com/tkarras/progressive-forming-of-gans. The Places2 dataset can be obtained from http://places2.csail.mit.edu/, and the Pconv random mask dataset is available at https://nv-adlr.github.io/publication/partialconv-inpainting.

## Declarations

**Conflict of interest** We declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Appendix A additional examples

In Figs. 15, 16 and 17, we show more examples on FFHQ [49] high-resolution face dataset images, which are occluded by central rectangular masks and random irregularities. Here,

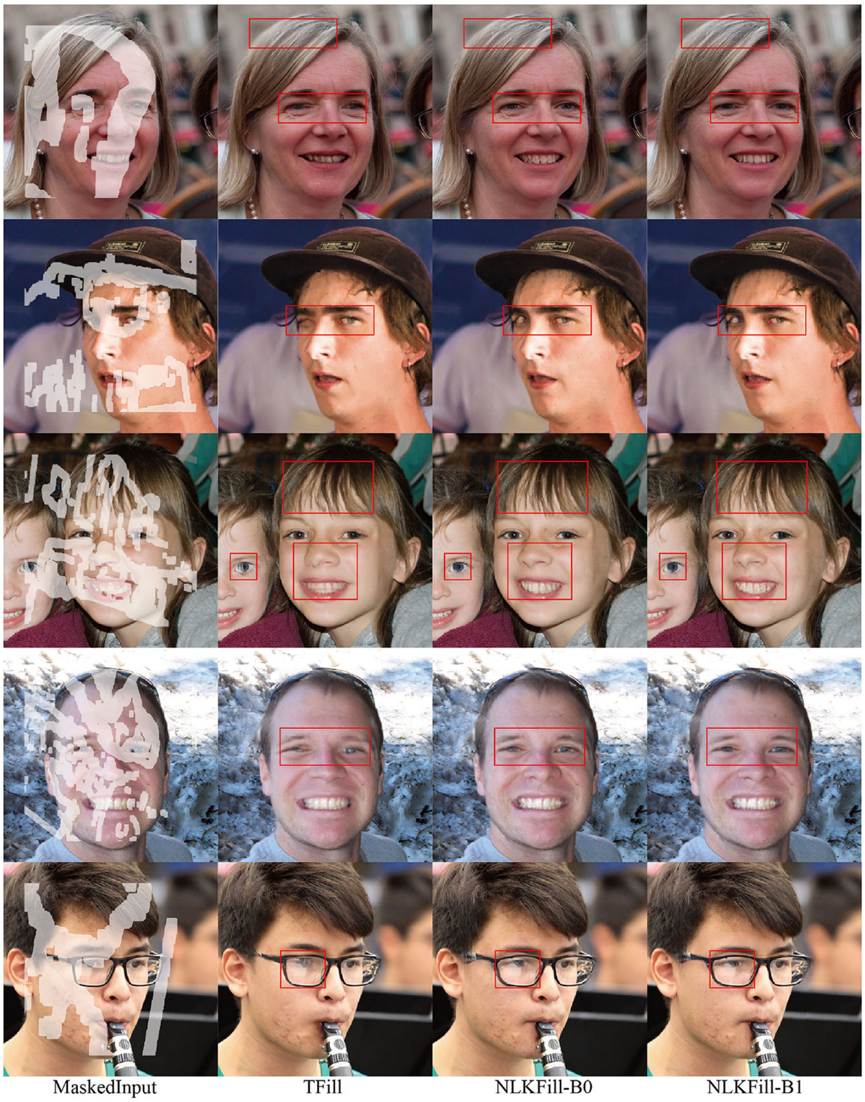


**Fig. 15** Additional results comparing with TFill. Additional results between TFill and NLKFill-B0 and NLKFill-B1 on the FFHQ test set. In the results, you can focus on the quality comparison of the red frame line position in the picture. In contrast, our models NLKFill-B0 and NLKFill-B1 are superior in both eye and hair restoration, and can generate a more realistic appearance

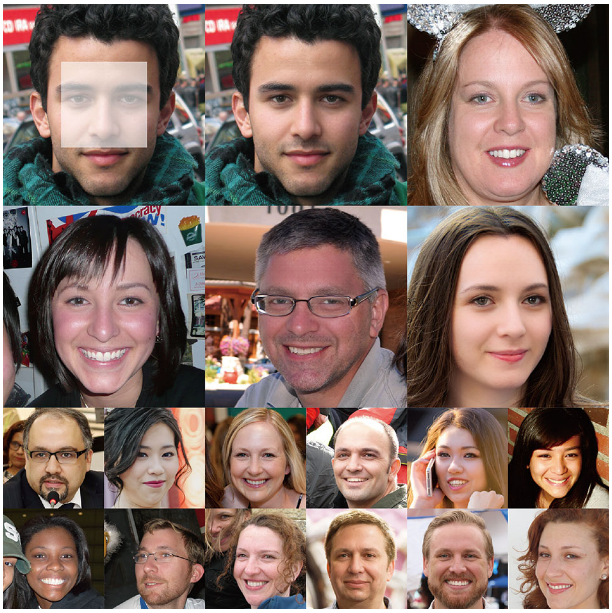


**Fig. 16** Additional results for NLKFill-B1. Example inpainting results of our method NLKFill-B1 on the FFHQ face dataset. Here, the center mask is used for all input images. An example input for center masking is shown on the upper left. As can be seen, the finished images are of high average quality. Even in some challenging cases, such as when glasses are covered by the center, our method can correctly inpaint faces with glasses. According to the image sequence, the corresponding image resolution is 512 × 512, 256 × 256, increasing from top to bottom

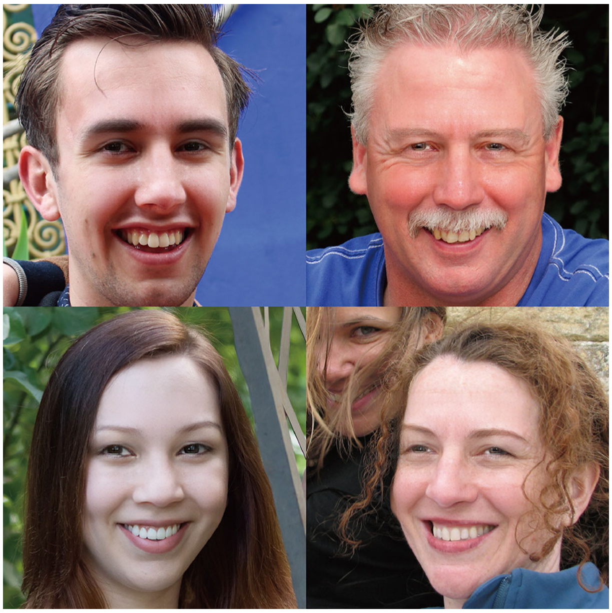


**Fig. 17** Additional high-resolution inpainting results for NLKFill-B1. Using the model NLKFill-B1 and the face dataset FFHQ to directly use high-definition face images with a resolution of 1024×1024 for training. The mask uses the same central rectangular mask as in Fig. 16. It can be seen from the results that NLKFill still captures enough detailed features from the high-resolution image, making the restored structure more refined and natural, and finally outputs a high-resolution restoration result

all examples shown are selected from the corresponding test set. In Fig. 15, we continue to supplement the comparative results of TFill on multiple test examples. In Figs. 16 and 17, we demonstrate the inpainting capabilities of NLKFill for center-corrupted face images at different resolutions. These examples nicely demonstrate that our model is suitable for multi-resolution inpainting tasks under different masks, and synthesizes semantically consistent content with a visually realistic appearance based on all existing visible pixels.

## References

1. Ballester C, Bertalmio M, Caselles V, Sapiro G, Verdera J (2001) Filling-in by joint interpolation of vector fields and gray levels. IEEE Trans Image Process 10(8):1200–1211
2. Barnes C, Shechtman E, Finkelstein A, Goldman DB (2009) Patchmatch: A randomized correspondence algorithm for structural image editing. ACM Trans Graph 28(3):24
3. Criminisi A, Pérez P, Toyama K (2004) Region filling and object removal by exemplar-based image inpainting. IEEE Trans Image Process 13(9):1200–1212
4. Iizuka S, Simo-Serra E, Ishikawa H (2017) Globally and locally consistent image completion. ACM Trans Graph (ToG) 36(4):1–14
5. Liu G, Reda FA, Shih KJ, Wang T-C, Tao A, Catanzaro B (2018) Image inpainting for irregular holes using partial convolutions. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 85–100
6. Liu H, Jiang B, Song Y, Huang W, Yang C (2020) Rethinking image inpainting via a mutual encoder-decoder with feature equalizations. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16, pp. 725–741. Springer
7. Nazeri K, Ng E, Joseph T, Qureshi F, Ebrahimi M (2019) Edgeconnect: Structure guided image inpainting using edge prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, pp. 0–0
8. Pathak D, Krahenbuhl P, Donahue J, Darrell T, Efros AA (2016) Context encoders: Feature learning by inpainting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2536–2544
9. Peng J, Liu D, Xu S, Li H (2021) Generating diverse structure for image inpainting with hierarchical vq-vae. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10775–10784
10. Wan Z, Zhang J, Chen D, Liao J (2021) High-fidelity pluralistic image completion with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4692–4701
11. Yi Z, Tang Q, Azizi S, Jang D, Xu Z (2020) Contextual residual aggregation for ultra high-resolution image inpainting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7508–7517
12. Yu J, Lin Z, Yang J, Shen X, Lu X, Huang TS (2018) Generative image inpainting with contextual attention. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5505–5514
13. Zeng Y, Lin Z, Lu H, Patel VM (2021) Cr-fill: Generative image inpainting with auxiliary contextual reconstruction. In: Proceed-

ings of the IEEE/CVF International Conference on Computer Vision, pp. 14164–14173

14. Zheng C, Cham T-J, Cai J (2019) Pluralistic image completion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1438–1447
15. Zheng C, Cham T-J, Cai J, Phung D (2022) Bridging global context interactions for high-fidelity image completion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11512–11522
16. Li Y, Liu S, Yang J, Yang M-H (2017) Generative face completion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3911–3919
17. Liu H, Jiang B, Xiao Y, Yang C (2019) Coherent semantic attention for image inpainting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4170–4179
18. Azulay A, Weiss Y (2018) Why do deep convolutional networks generalize so poorly to small image transformations? arXiv preprint arXiv:1805.12177
19. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. Adv Neural Inform Process Syst 30
20. Chen M, Radford A, Child R, Wu J, Jun H, Luan D, Sutskever I (2020) Generative pretraining from pixels. In: International Conference on Machine Learning, pp. 1691–1703. PMLR
21. Parmar N, Vaswani A, Uszkoreit J, Kaiser L, Shazeer N, Ku A, Tran D (2018) Image transformer. In: International Conference on Machine Learning, pp. 4055–4064. PMLR
22. Guo M-H, Lu C-Z, Liu Z-N, Cheng M-M, Hu S-M (2022) Visual attention network. arXiv preprint arXiv:2202.09741
23. Bertalmio M, Vese L, Sapiro G, Osher S (2003) Simultaneous structure and texture image inpainting. IEEE Trans Image Process 12(8):882–889
24. Levin A, Zomet A, Weiss Y (2003) Learning how to inpaint from global image statistics. In: ICCV, vol. 1, pp. 305–312
25. Jia J, Tang C-K (2004) Inference of segmented color and texture description by tensor voting. IEEE Trans Pattern Anal Mach Intellig 26(6):771–786
26. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2020) Generative adversarial networks. Commun ACM 63(11):139–144
27. Mirza M, Osindero S (2014) Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784
28. Kingma DP, Welling M (2013) Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114
29. Yu J, Lin Z, Yang J, Shen X, Lu X, Huang TS (2019) Free-form image inpainting with gated convolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4471–4480
30. Portenier T, Hu Q, Szabo A, Bigdeli SA, Favaro P, Zwicker M (2018) Faceshop: Deep sketch-based face image editing. arXiv preprint arXiv:1804.08972
31. Jo Y, Park J (2019) Sc-fegan: Face editing generative adversarial network with user's sketch and color. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1745–1753
32. Cao C, Fu Y (2021) Learning a sketch tensor space for image inpainting of man-made scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 14509–14518
33. Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S (2020) End-to-end object detection with transformers. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16, pp. 213–229. Springer
34. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S et al (2020) An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929
35. Guo M-H, Xu T-X, Liu J-J, Liu Z-N, Jiang P-T, Mu T-J, Zhang S-H, Martin RR, Cheng M-M, Hu S-M (2022) Attention mechanisms in computer vision: A survey. Comput Visual Media 8(3):331–368
36. Chen L, Zhang H, Xiao J, Nie L, Shao J, Liu W, Chua T-S (2017) Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5659–5667
37. Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141
38. Woo S, Park J, Lee J-Y, Kweon IS (2018) Cbam: Convolutional block attention module. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 3–19
39. Wang Q, Wu B, Zhu P, Li P, Zuo W, Hu Q (2020) Eca-net: Efficient channel attention for deep convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11534–11542
40. Qin Z, Zhang P, Wu F, Li X (2021) Fcanet: Frequency channel attention networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 783–792
41. Yuan Y, Huang L, Guo J, Zhang C, Chen X, Wang J (2018) Ocnet: Object context network for scene parsing. arXiv preprint arXiv:1809.00916
42. Zhang H, Goodfellow I, Metaxas D, Odena A (2019) Self-attention generative adversarial networks. In: International Conference on Machine Learning, pp. 7354–7363. PMLR
43. Wang F, Jiang M, Qian C, Yang S, Li C, Zhang H, Wang X, Tang X (2017) Residual attention network for image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3156–3164
44. Hu J, Shen L, Albanie S, Sun G, Vedaldi A (2018) Gather-excite: Exploiting feature context in convolutional neural networks. Adv Neur Inform Process Syst 31
45. Park J, Woo S, Lee J-Y, Kweon IS (2018) Bam: Bottleneck attention module. arXiv preprint arXiv:1807.06514
46. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778
47. Hendrycks D, Gimpel K (2016) Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415
48. Esser P, Rombach R, Ommer B (2021) Taming transformers for high-resolution image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12873–12883
49. Karras T, Laine S, Aila T (2019) A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4401–4410
50. Zhang B, Gu S, Zhang B, Bao J, Chen D, Wen F, Wang Y, Guo B (2022) Styleswin: Transformer-based gan for high-resolution image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11304–11314
51. Gal R, Hochberg DC, Bermano A, Cohen-Or D (2021) Swagan: a style-based wavelet-driven generative model. ACM Trans Graph (TOG) 40(4):1–11
52. Zhao H, Gallo O, Frosio I, Kautz J (2016) Loss functions for image restoration with neural networks. IEEE Trans Comput Imag 3(1):47–57
53. Johnson J, Alahi A, Fei-Fei L (2016) Perceptual losses for real-time style transfer and super-resolution. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14, pp. 694–711. Springer

54. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556
55. Liu Z, Luo P, Wang X, Tang X (2018) Large-scale celebfaces attributes (celeba) dataset. Retrieved August 14: 11
56. Zhou B, Lapedriza A, Khosla A, Oliva A, Torralba A (2017) Places: a 10 million image database for scene recognition. IEEE Trans Pattern Anal Mach Intellig 40(6):1452–1464
57. Zhang R, Isola P, Efros AA, Shechtman E, Wang O (2018) The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 586–595
58. Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S (2017) Gans trained by a two time-scale update rule converge to a local nash equilibrium. Adv Neur Inform Process Syst 30

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.