



Discrete codebook collaborating with transformer for thangka image inpainting

Jinxian Bai¹ · Yao Fan¹ · Zhiwei Zhao¹

Received: 14 June 2024 / Accepted: 30 July 2024

© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2024

Abstract

Thangka, as a precious heritage of painting art, holds irreplaceable research value due to its richness in Tibetan history, religious beliefs, and folk culture. However, it is susceptible to partial damage and form distortion due to natural erosion or inadequate conservation measures. Given the complexity of textures and rich semantics in thangka images, existing image inpainting methods struggle to recover their original artistic style and intricate details. In this paper, we propose a novel approach combining discrete codebook learning with a transformer for image inpainting, tailored specifically for thangka images. In the codebook learning stage, we design an improved network framework based on vector quantization (VQ) codebooks to discretely encode intermediate features of input images, yielding a context-rich discrete codebook. The second phase introduces a parallel transformer module based on a cross-shaped window, which efficiently predicts the index combinations for missing regions under limited computational cost. Furthermore, we devise a multi-scale feature guidance module that progressively fuses features from intact areas with textural features from the codebook, thereby enhancing the preservation of local details in non-damaged regions. We validate the efficacy of our method through qualitative and quantitative experiments on datasets including Celeba-HQ, Places2, and a custom thangka dataset. Experimental results demonstrate that compared to previous methods, our approach successfully reconstructs images with more complete structural information and clearer textural details.

Keywords Image inpainting · Thangka images · Transformer · Cross-shaped window attention · Codebook

1 Introduction

Thangka images originate from the Tibetan region as a traditional religious painting art form, presented as scroll paintings adorned with brocades. Locals typically employ precious minerals like gold, silver, pearls, and agate, along with plant-based pigments, to paint on cotton or silk. The content of thangkas reflects the social life, customs, myths, legends, and historical events of the Tibetan people, earning

them the title of the “Tibetan Encyclopedia.” However, during creation, transmission, or storage, thangka images often suffer local damage or degradation due to harsh natural conditions and human activities, significantly impairing visual appreciation and subsequent research. Traditional physical inpainting is time-consuming and irreversible, hindering the inheritance and preservation of cultural heritage.

With the flourishing development of scientific research and computer technology, digital methods enable the inpainting of images in a non-destructive manner [1–4], significantly enhancing the science and reliability of inpainting work. Digital image inpainting is a crucial research direction in computer vision, aiming to repair damaged parts of pixel features in incomplete images using mathematical models and machine learning techniques, subsequently reconstructing and generating high-quality images akin to the original in deep semantic content. Early inpainting models based on Convolutional Neural Networks (CNNs) [5–8], incorporating encoder-decoder architectures or Generative Adversarial Networks (GANs), handle large volumes of spatially

Communicated by Yongdong Zhang.

✉ Yao Fan
fannyaoo@xzmu.edu.cn

Jinxian Bai
1956323825@qq.com

Zhiwei Zhao
1041653051@qq.com

¹ School of Information Engineering, Xizang Minzu University, Xianyang 712000, Shaanxi, China

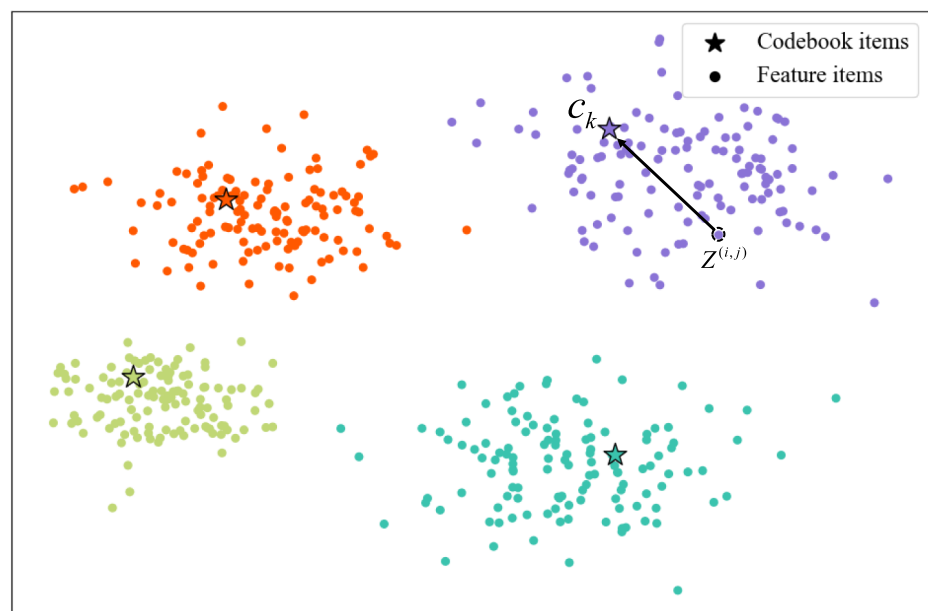
correlated data to generate new pixel content for damaged areas. Moreover, some approaches [9, 10] incorporate attention mechanisms, ensuring neighboring pixels focus on similar background regions, thus enhancing the coherence and authenticity of inpainting outcomes. Nonetheless, these methods lack emphasis on prior information such as structural textures, leading to semantically implausible and blurry textures in restored results. Some researchers have proposed utilizing auxiliary information for structural recovery, including contours [11], edges [12–14], and semantics [15, 16]. These methods are suited for highly structured images like faces or architecture. Unlike natural images, however, thangka feature intricate and exquisite patterns in the attire of deities, ritual objects, or the florid, cloudy, and landscape backgrounds, brimming with detail. Moreover, Thangka images are more focused on religious and cultural symbols, which demands that image restoration algorithms not only restore physical integrity but also preserve artistic value and cultural significance. When confronted with such thangka images, the accuracy of predicted prior information decreases dramatically, resulting in extensive inpainting errors during the filling of internal textures.

Inspired by iGPT's [17] remarkable performance in various visual tasks [18–20], Wan et al. [21] adopted K-means clustering to generate a visual vocabulary, mapping low-resolution images to indices in this vocabulary to form a discrete representation, thereby facilitating the use of Transformers [22] to restore the coherent structure and coarse texture of images. However, fundamental issues with the Transformer, such as high computational complexity and inadequate handling of positional information, persist. Recently, Patrick Esser et al. [23] introduced VQGAN, a

novel method that first employs a Convolutional Neural Network (CNN) to encode images into continuous latent representations, which are subsequently transformed into discrete representations based on a codebook via vector quantization. This is followed by efficient global modeling of the codebook indices in high-resolution images using Transformers. Here, as depicted in Fig. 1, the codebook can be seen as a form of prior knowledge, learned from extensive training data through clustering algorithms. It encompasses multiple representative codebook items, which can be regarded as compact representations of various typical patterns or structures found in images. Consequently, recent discrete codebook-based generative models [24–26], with their capability for discrete image representation and heightened robustness against various degradation scenarios, have found broad application in image super-resolution. Nonetheless, exploration of the quantized discrete codebook mechanism in the realm of image inpainting [27, 28] remains limited. Furthermore, this solution is beset by three salient issues: (1) significant information loss occurs during the conversion of original images into continuous latent representations, leading to color distortion and artifact issues in reconstructions; (2) fidelity of non-masked regions in repaired images is low, characterized by poor quality and insufficient detail; (3) the attention scope of the transformer modules is constrained, resulting in suboptimal predictive performance and substantial computational resource consumption.

To address the aforementioned issues, this paper proposes a novel framework for image inpainting that synergistically combines a discrete codebook with a transformer (CDCT). Diverging from previous methods based on structural priors, our approach integrates the advantages of codebook-based

Fig. 1 Illustration of the distribution of latent features and their codebook entries



methods and Vector Quantization (VQ) techniques [23, 29], efficiently representing image information. The key contributions are outlined as follows:

- (1) We devise a novel codebook learning framework where an encoder partitions the input image into fixed-size, non-overlapping patches, followed by nonlinear transformations into latent feature vectors. This ensures effective isolation of local information. The integration of a vector quantization codebook further enhances the capture and retention of image structural information and details, yielding more realistic outcomes during reconstruction.
- (2) A parallel Cross-Shaped Window (CSWin) Transformer module is designed, featuring a cross-shaped window and locally enhanced positional encoding. This design bolsters context modeling capabilities while reducing computational costs, thereby enhancing the accuracy of index prediction.
- (3) Innovatively, we introduce a multi-scale feature guidance module, wherein Large Kernel Attention (LKA) at different scales leverages local information and adaptivity across channels to better learn feature representations of non-damaged areas.
- (4) Extensive experimentation with our CDCT model has been conducted on datasets including Celeba-HQ, Places2, and a custom Thangka dataset, comparing it against state-of-the-art methodologies. Both qualitative and quantitative assessments demonstrate the competitiveness of the CDCT model's inpainting results. It opens up a novel technical path for the inpainting of cultural heritage such as thangka images.

2 Related work

2.1 Image inpainting

Image inpainting research can be categorized into two classes: traditional methods that fill in missing parts based on diffusion [1, 2] or sample-based approaches [3], and deep learning methodologies that learn semantic and textural structures from vast amounts of image data. Diffusion-based inpainting methods typically rely on mathematical diffusion equations, simulating the propagation process of image signals in two-dimensional or higher-dimensional spaces to achieve the goal of filling in image gaps. These methods can lead to detail loss or boundary artifacts when dealing with nonlinear degradations or sharp edge variations. Sample-based repair techniques involve selecting suitable sample textures or structures from undamaged areas of the image or other reference images with similar features, to replicate or blend them to reconstruct damaged regions. These

methods tend to produce matching errors or edge discontinuities when confronted with missing foreground regions possessing complex textures and structures. The remarkable achievements of deep learning techniques in tasks such as image classification and object detection have sparked the interest of researchers in applying these methodologies to the task of image inpainting [30]. Cui et al. [31] proposed a multi-branch and content-aware module that dynamically decomposes features into independent frequency subbands and highlights useful information through channel attention weights. Wang et al. [7] designed a multi-column network architecture comprised of multiple parallel encoder-decoder branches, where the structure synthesizes different components of the image in parallel within a single stage, enhancing the quality of image repair. Sagong et al. [10] constructed a shared encoding network accompanied by two parallel decoding paths. A coarse path rapidly generates an initial repair outcome, while the refinement path further enhances repair quality atop this foundation, leveraging contextual attention mechanisms. Guo et al. [12] introduced a Conditional Texture and Structure Dual Generative Network, incorporating a bidirectional gated feature fusion module for the exchange and integration of information across structural and textural domains, as well as a contextual feature aggregation module that utilizes regional affinity learning for further refinement of generated content. Nonetheless, these approaches tend to yield over-smoothed or texture-discontinuous results when restoring complex textures and fine structures. FocalNet [32] emphasizes critical information and enhances the network's capability to process high-resolution features through a dual-domain selection mechanism and multi-scale feature splitting. Consequently, the feature guidance module designed in our method is not only capable of capturing global contextual information but also meticulously handles local details, thereby promoting comprehensive circulation and utilization of information within the model.

2.2 Vision transformer

Recently, Dosovitskiy et al. proposed an image classification model known as Vision Transformer (ViT) [33]. This model aims to introduce the Transformer architecture, which originally shone in the field of Natural Language Processing (NLP), to computer vision tasks, especially image classification. Until then, Convolutional Neural Networks had been the dominant models in visual tasks. The advent of ViT highlighted the potential of Transformers in image processing domains. Subsequently, new variants of Transformers, including the Axial Transformer [34] and the CSWin Transformer [35], have continually emerged and shown promise in the domain of image inpainting. Notable examples are the ZITS model proposed by Dong et al. [13], and the TFill model proposed

by Zheng et al. [36]. In addressing the repair of large missing areas, Transformers, thanks to their unique self-attention mechanism, are capable of generating realistic textures. Compared to conventional convolution operations, Transformers boast a wider receptive field but entail a higher computational overhead. Hence, we have made adjustments to the standard Transformer architecture by incorporating cross-shaped windows to segment feature maps, with the dual purpose of reducing computational demands while enhancing repair efficacy.

2.3 Vector-quantized codebook

The concept of vector quantized codebooks was initially introduced in the groundbreaking work of VQVAE [29], marking a significant innovation in the field of generative models. Within this framework, the encoder's output deviates from conventional continuous representations, instead adopting a discretized form achieved through lookup in a meticulously designed codebook. Remarkably, this codebook is not pre-determined but dynamically optimized as part of the model learning process, thereby encapsulating a profound understanding of the data distribution. Building upon this foundation, subsequent research has relentlessly pursued the refinement and expansion of codebook learning mechanisms. VQVAE2 [37] advanced this by incorporating a hierarchical codebook structure, enabling the model to capture different scales of image features at a finer granularity. Following this, VQGAN aimed to generate high-quality images, enhancing the realism and diversity of generated samples while maintaining the efficient encoding of VQVAE achieved through adversarial training with GANs. Moreover, other works have explored applying VQ codebooks to visual tasks such as super-resolution and image editing, yielding remarkable achievements. Zhu et al. [24] proposed an asymmetric VQGAN architecture that effectively addresses information loss issues in the original VQGAN when applied to StableDiffusion, thereby enhancing the quality of image editing and generation. Chen et al. [38] leveraged the implicitly learned high-resolution prior from a pre-trained VQGAN network, framing super-resolution as a feature matching problem between low-resolution features and a codebook of distortion-free high-resolution feature. The method performs well in recovering fine details such as real hairs. Inspired by codebook learning, this paper explores a novel method for Thangka inpainting based on vector quantized codebooks.

3 Overall design of the model structure

Thangka images not only exhibit rich and unique textural and structural features but also boast visually striking effects with saturated colors and pronounced contrasts. In the face

of degraded images resulting from masking processes, this paper aims to train a model that can fully exploit and effectively utilize the abundant textural and complex structural information in thangkas to infer logically consistent content for missing regions and supplement visually realistic details, thereby achieving the complete inpainting of the image.

To attain this objective, we propose a discrete codebook and transformer collaborative image inpainting model, whose overall framework is depicted in Fig. 2. Built upon the fundamental framework of GANs, the training procedure encompasses two primary phases: a shared codebook learning phase and an image inpainting phase based on codebook priors. During the shared codebook learning phase, an encoder-decoder structure is employed to jointly learn a discrete codebook representing the input image. Inspired by the work [36], before vector quantization, we enhance the encoder architecture to map the input image into a continuous latent space representation via non-overlapping patches. This ensures the integrity of information between patches and prevents erroneous information transmission between masked and unmasked areas in the subsequent phase. In the second phase, leveraging the constructed codebook priors, we utilize a designed parallel CSWin Transformer module to accurately predict the indices of missing tokens. Subsequently, these indices are used to retrieve the corresponding discrete vectors from the codebook. This module is capable of generating high-quality repair outcomes while maintaining a balance in computational complexity. Furthermore, we propose a multi-scale feature guidance module to fully leverage features from non-damaged regions, promoting greater harmony and consistency in structure and texture between the generated and intact areas, thereby enhancing the quality and fidelity of the repair results.

3.1 Discrete codebook learning phase

3.1.1 Codebook learning

During the codebook learning phase, the model architecture comprises three core components: the codebook encoder E , the codebook decoder G , and a codebook $\mathcal{C} = \{c_k\}_{k=0}^K$ containing K discrete codewords. When processing the input image $I_t \in \mathbb{R}^{H \times W \times 3}$, the codebook encoder E initially comes into play, transforming image I into a latent representation Z in a high-dimensional space, i.e., $Z = E(x) \in \mathbb{R}^{m \times n \times d}$, where d denotes the number of dimensions making up this latent vector. Subsequently, an element-wise quantization operation $q(\cdot)$ is employed, comparing each element in the spatial latent representation Z with all codewords in the codebook \mathcal{C} . The codeword C_k that is closest in distance is identified, and the index of this codeword in the codebook (i.e., the code token sequence $s, s \in \{0, \dots, N-1\}^{m \times n}$) is recorded. This forms a code token sequence of size $m \times n$, which is

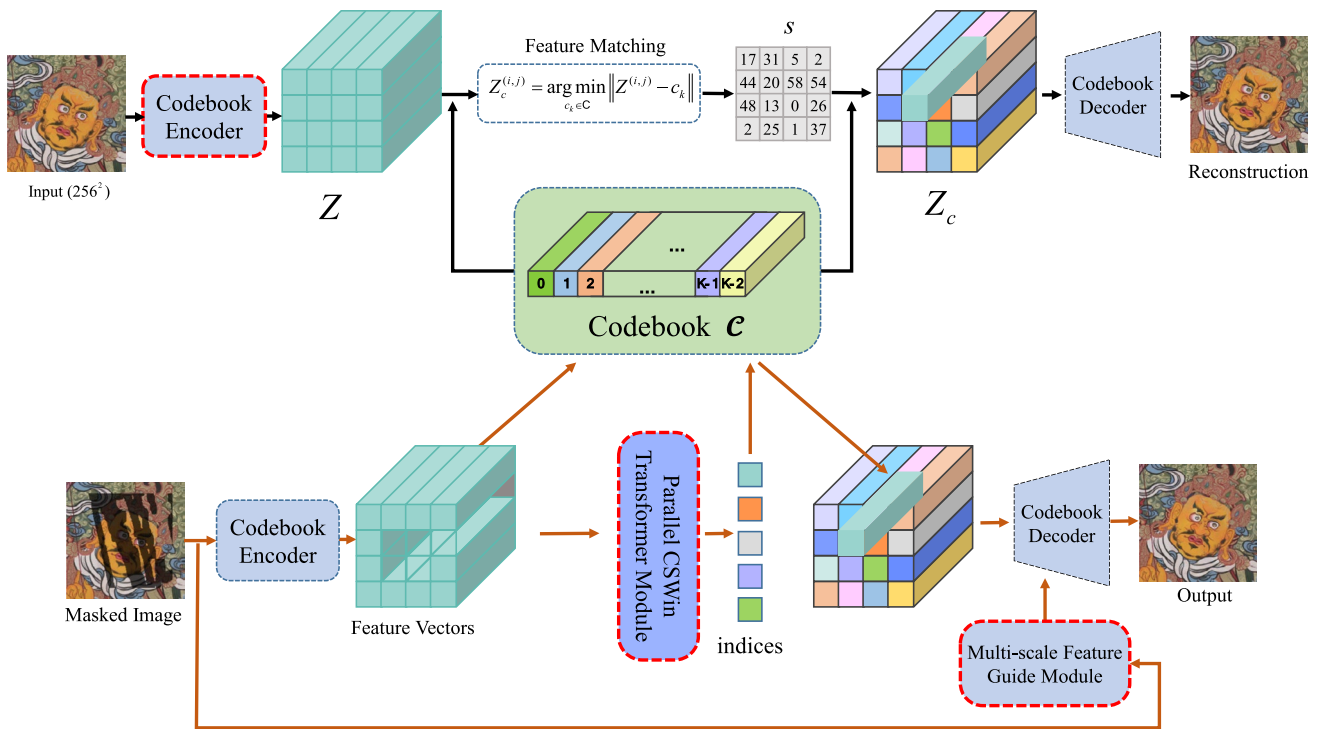


Fig. 2 Overall framework of the CDCT model

used to retrieve corresponding code items from the codebook \mathcal{C} to constitute the quantized feature Z_c .

$$\begin{aligned}
 Z_c^{(i,j)} &= \mathbf{q}(Z) := \arg \min_{c_k \in \mathcal{C}} \|Z^{(i,j)} - c_k\|; \\
 s^{(i,j)} &= \arg \min_k \|Z^{(i,j)} - c_k\|_2.
 \end{aligned}
 \tag{1}$$

Specifically, if $s^{(i,j)} = k$, then the value of Z_c at position (i,j) is equal to the k th code item \mathcal{C}_k in the codebook. Subsequently, the decoder G reconstructs a high-quality image I_{rec} given Z_c . The overall reconstructed image $I_{rec} \approx I_t$ can be formulated as:

$$I_{rec} = G(Z_c) = G(\mathbf{q}(E(I))).
 \tag{2}$$

The encoder performs a mapping operation that transforms image data of size $H \times W$ into a discrete encoded form of scale $H/m \times W/n$, where parameters m and n denote the downsampling ratio. This process essentially condenses the information within each $m \times n$ region of the image I_t into a single encoding unit. Thus, when referring to any encoding element in Z_c , the $m \times n$ equally symbolizes the corresponding coverage of that code on the original image I_t space.

For end-to-end training of the codebook and model, four image-level reconstruction losses are employed: L1 loss \mathcal{L}_1 , perceptual loss \mathcal{L}_{per} [39], adversarial loss \mathcal{L}_{adv} [40], and style loss \mathcal{L}_{style} [41]. The specific definitions of these loss functions are as follows:

$$\begin{aligned}
 \mathcal{L}_1 &= \|I_t - I_{rec}\|_1; \quad \mathcal{L}_{per} = \|\Phi(I_t) - \Phi(I_{rec})\|_2^2; \\
 \mathcal{L}_{adv} &= [\log D(I_t) + \log(1 - D(I_{rec}))]; \\
 \mathcal{L}_{style} &= \mathbb{E} \left[\sum_k \frac{1}{M_k} \|G_{\varphi_k}(I_t) - G_{\varphi_k}(I_{rec})\|_1 \right].
 \end{aligned}
 \tag{3}$$

Here, Φ denotes the feature extractor from the VGG19 network [42]. Due to the inadequacy of image-level loss constraints when updating codebook entries, an additional intermediate code-level loss $\mathcal{L}_{quantize}$ is adopted in this work to minimize the discrepancy between the codebook \mathcal{C} and the embedded input features Z .

$$\mathcal{L}_{quantize} = \|sg(Z) - Z_c\|_2^2 + \beta \|Z - sg(Z_c)\|_2^2,
 \tag{4}$$

Here, $sg(\cdot)$ refers to the stop-gradient operator, and the parameter β is set to 0.25 to balance the weights between the encoder and codebook update speeds. Addressing the non-differentiability issue inherent in the feature quantization process as shown in Eq. (1), we adopt the approach presented in reference [43], employing a straight-through strategy. This strategy entails mirroring gradients from the decoding stage to the encoding stage during backpropagation, ensuring the feasibility of the backpropagation process. To comprehensively guide the learning of codebook priors, the combined loss function $\mathcal{L}_{codebook}$ serves as the

optimization objective, driving the entire end-to-end training process. The $\mathcal{L}_{codebook}$ is defined as:

$$\mathcal{L}_{codebook} = \mathcal{L}_1 + \mathcal{L}_{per} + \mathcal{L}_{quantize} + \lambda_{adv} \cdot \mathcal{L}_{adv} + \mathcal{L}_{style}, \quad (5)$$

where λ_{adv} is set to 0.8 in the experiments in this paper.

While a larger codebook size could simplify reconstruction, redundant elements may lead to ambiguity in subsequent code prediction. Consequently, our CDCT method sets the number of codebook entries N to 1024, which is sufficient for accurate image reconstruction. Moreover, the codebook dimension d is set to 256.

3.1.2 Design of the encoder-decoder

Traditional CNN-based encoders, which process input images using a sliding window approach with several convolutional kernels, are ill-suited for image inpainting tasks as they tend to introduce interference between masked and unmasked regions. Consequently, the encoder in the first phase of our model is designed to handle the input image via non-overlapping patches through a series of linear residual layers. Specifically, our Token representations are extracted using a linear residual structure across eight blocks. Within each block, two sets of operations comprising GELU activation functions, linear layers, and residual connections are employed. The input image is initially unfolded into patches, altering its dimensions to $(3 \times m \times n, L)$, where L denotes the number of patches. This is followed by an adaptation layer that transforms the features into dimensions of (L, d) . Thereafter, within each block, the input features transform 256 dimensions to 128, and then back to 256 dimensions. After the feature extraction through these eight linear residual layers, a fold operation is applied to obtain the latent representation Z . Consequently, a substantial compression ratio $r = H/n = W/m = 32$ is achieved, which confers robustness against degradation in the global modeling of the second phase while maintaining computationally manageable costs.

The decoder G of this paper comprises three transposed convolutional layers followed by one standard convolutional layer, all serving the purpose of upsampling. Here, the transposed convolution kernels are of size 4×4 , indicating both their width and height are 4 pixels. A stride of 2 is adopted, meaning the kernel moves 2 pixels at a time across the input image. The padding size is set to 1, signifying a single pixel padding is added to the edges of the input to maintain the spatial dimensions of the output. Initially, three transposed convolutions upscale features from $256 \times 32 \times 32$ to $64 \times 256 \times 256$ dimensions. Subsequently, a convolutional layer with a 3×3 kernel,

reflection padding of 1, and a stride of 1 adjusts the output to a final resolution of $256 \times 256 \times 3$, yielding the reconstructed image.

3.2 Image inpainting stage with codebook priors

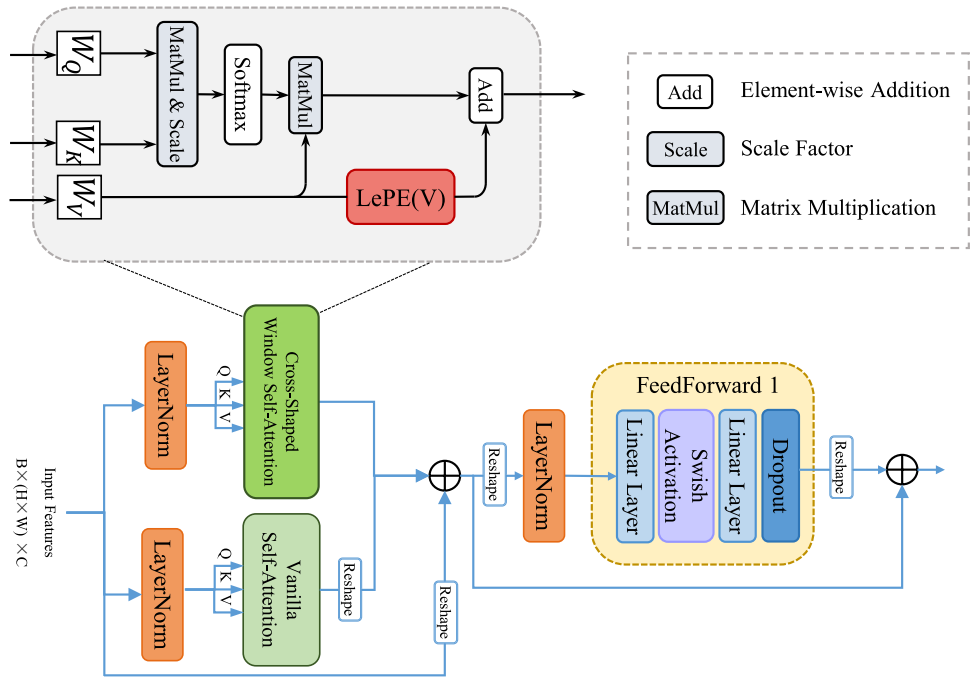
3.2.1 Parallel CSWin transformer modules

In existing transformer architectures employed for image inpainting and completion, the indices of quantized pixels serve as both inputs and prediction targets. While this strategy of leveraging contextual indices to predict missing indices enhances computational efficiency, it poses a significant issue of information loss in the input domain, which is detrimental to index sequence prediction. Consequently, our proposed Parallel CSWin Transformer module (PCT) takes the feature vectors \hat{Z} from the codebook encoder as direct inputs, facilitating more accurate predictions while mitigating information loss.

The parallel CSWin transformer module is shown in Fig. 3. We first add additional learnable positional embeddings to the feature vector $\hat{Z} \in \mathbb{R}^{(H \times W) \times C}$ to preserve spatial information and subsequently flatten the feature vector along the spatial dimension to obtain the final input to the module. Our model employs 12 parallel CSWin transformer blocks, each comprising parallel Multi-Head Self-Attention blocks and cross-shaped window attention [35] blocks, along with a Feedforward layer (Feedforward 1). The number of attention heads is set to 8. Departing from conventional transformer modules, the PCT module combines multi-head and cross-shaped windowing, significantly reducing computational overhead while achieving superior repair outcomes. Moreover, the cross-shaped window attention block introduces a novel positional encoding mechanism, LePE, on the linearly projected values (V), enhancing local inductive biases. Notably, within the PCT module, the cross-shaped window attention and full self-attention operate on different receptive fields during training and then are concatenated via residual connections, thereby ensuring that standard self-attention blocks remain unaffected by the CSWin attention blocks. The Swish activation function in Feedforward layer 1 smoothens gradients while preserving the nonlinear characteristics of ReLU.

Unlike axial attention [34], the cross-shaped window attention splits channels into horizontal and vertical stripes, with half of the heads capturing attention along horizontal stripes and the other half along vertical stripes. Specifically, focusing on horizontal stripe self-attention as an example, the feature matrix S is evenly divided into a series of non-overlapping horizontal stripe segments $[S^1, \dots, S^N]$, where $N = H/b$, and each segment comprises b columns and W rows of elements. Furthermore, the hyperparameter b can be

Fig. 3 Parallel CSWin Transformer module



adjusted flexibly to strike a balance between learning capacity and computational expense. Assuming the dimension of query, key, and value vectors for each head is d , the output of horizontal stripe self-attention processed by each head can be defined by the following expression:

$$\begin{aligned}
 S &= [S^1, S^2, \dots, S^N], \\
 Y^i &= \text{Attention}(S^i W^Q, S^i W^K, S^i W^V), \\
 \text{Attention}^H(S) &= [Y^1, Y^2, \dots, Y^N]
 \end{aligned}
 \tag{6}$$

where $S^i \in \mathbf{R}^{(b \times W) \times C}$, $i = 1, \dots, N$. Here, $W^Q \in \mathbf{R}^{C \times d}$, $W^K \in \mathbf{R}^{C \times d}$, and $W^V \in \mathbf{R}^{C \times d}$ denote the query, key, and value matrices respectively, obtained after linear transformations of the input feature matrix for each head. Analogously, local self-attention operations applied to the vertical stripe regions can be derived accordingly, with outputs from each head represented by $\text{Attention}^V(S)$.

The output from the final PCT block is further projected into a probability distribution over K potential vectors in the codebook, using a linear layer followed by a Softmax function, which represents the likelihood of image patches corresponding to features in the codebook. To quantify the agreement between the model’s predictions and class labels, the PCT module is trained to predict the probability distribution $p(s_i | s_{<i})$ of the next likely index. This sets the training objective as minimizing the negative log-likelihood of the data representation:

$$L_{\text{Transformer}} = \mathbb{E}_x p(x) [-\log p(s)].
 \tag{7}$$

where $p(s) = \prod_i p(s_i | s_{<i})$.

3.2.2 Multi-scale feature guidance module

As illustrated in Fig. 4, we have devised a multi-scale feature guidance module aimed at preserving details in non-masked regions of the image. Given an input image that is masked with a mask m , yielding the masked input Y , this module represents the masked image input across multiple feature maps rather than compressing it into a single-layer feature.

Recognizing the need for comprehensive learning of multi-scale feature information with a large receptive field, we note that while the frequency-domain filtering convolution used in the LaMa approach [44] possesses a global receptive field, it is limited in capturing stable feature correlations between masked and unmasked regions. Meanwhile, numerous inpainting techniques incorporate diverse attention mechanisms to address long-range dependencies, but these attention-based strategies, lacking scale invariance, can lead to overfitting issues at specific resolutions. To address these challenges, we innovatively introduce large-kernel convolutions within our multi-scale feature guidance module, aiming to combine the strengths of CNN operations and attention mechanisms. This module is comprised of three LKA feature transformation sub-modules, each operating at a different scale. Specifically, we employ the Large Kernel Attention (LKA) structure proposed by Guo et al. [45], which utilizes $\frac{k}{d} \times \frac{k}{d}$ depthwise convolutions (DW-Conv) with dilation rate d to extract local features, followed by a $(2d - 1) \times (2d - 1)$ dilated depthwise convolution (DW-D-Conv) to capture long-range dependencies. Lastly, information is aggregated and channel numbers are adjusted through pointwise 1×1 convolutions, enhancing inter-channel interactions. As LKA

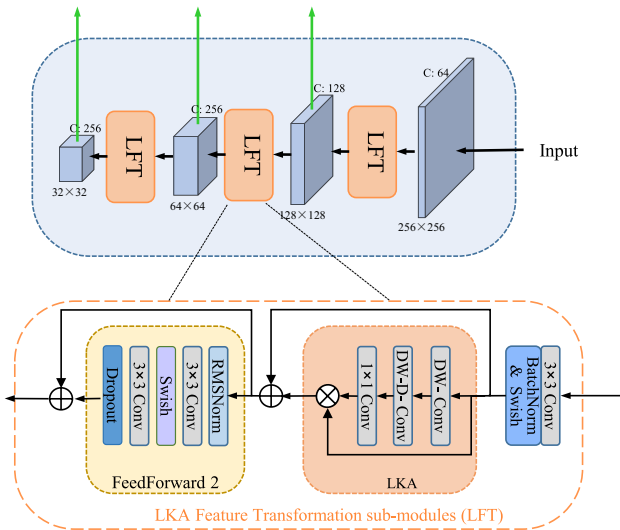


Fig. 4 Illustration of the multi-scale feature guidance module

focuses on optimizing feature representations in occluded areas with an expansive receptive field, it aids in the global learning of regular textures in the frequency domain. Furthermore, to ensure the generalizability of LKA, we append a Feedforward Network 2 (FFN2) after the LKA module. Our designed FFN2 comprises RMS normalization, 3×3 convolutions, Swish activations, another 3×3 convolution, and Dropout. We employ RMSNorm [46] for normalization to enhance training stability; the Swish function [47] is used to provide smoother gradients compared to ReLU while maintaining non-linearity, addressing the zero-centered gradient issue of ReLU for negative inputs.

4 Experimental design and results analysis

Experiments were conducted on three distinct datasets to train and evaluate our model: Celeba-HQ [48], an extended version of the CelebA dataset featuring high-quality, high-resolution facial images, from which we selected 27,000 images for training and 3000 for testing and validating the model’s performance on high-fidelity facial images. Places2 [49], a large-scale natural scene image dataset, saw us utilizing 20 scene categories for our experiments, with 90,000 images dedicated to training and 10,000 for evaluating the model’s performance in various scene understanding contexts. A custom Tibetan thangka dataset was also included, comprising buddhist thangka, dharmapala thangka, exoteric and esoteric thangka, and domestic thangka. What distinguishes this dataset is its highly detailed patterns, vivid colors, and themes related to religion and culture. This dataset featured 2500 images for training and 500 for testing and validation. Utilizing our custom Thangka image dataset

allows for a more accurate evaluation of the model’s ability to handle images with specific artistic styles and cultural elements. The specific configurations for the three datasets are outlined in Table 1. Irregular masks provided by PConv [50] were employed for both training and testing purposes.

For quantitative comparisons, this study employs a variety of image quality metrics, encompassing traditional measures such as Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), Mean Absolute Error (MAE), as well as the more recent, perceptually driven Learning Perceptual Image Patch Similarity (LPIPS).

4.1 Implementation details

For the first stage of training, our method employs the Adam optimizer ($\beta_1 = 0, \beta_2 = 0.9$) with a batch size of 16. The second stage also utilizes the Adam optimizer ($\beta_1 = 0.9, \beta_2 = 0.95$) but with a reduced batch size of 4. Learning rates are set to $2e-4$ and $3e-4$ for the two stages respectively, and a cosine scheduler is adopted for learning rate decay. Our method is implemented using the PyTorch framework and trained on a single NVIDIA 3090 GPU.

In this section, the proposed model is compared with other advanced methods, including EC [51], CTSDG [12], ICT [21], PUT [27], and MAT [52]. All comparative models are evaluated on both the Celeba-HQ and Places2 datasets. Additionally, we retrain EC, CTSDG, ICT, PUT, and MAT on our custom thangka dataset to further discuss the inpainting effects.

As illustrated in Fig. 5, the training progress of the first-stage network on the Places2 dataset is demonstrated. During training, the Quantize loss and Adv loss initially rise and then stabilize after fluctuating. On the right, through continuous training and tuning, the L1 loss, Perceptual loss, and Style loss of our model gradually decrease, contributing to an enhancement in the quality of generated images.

4.2 Qualitative comparisons

Figures 6, 7, and 8 present visual comparisons of the inpainting results achieved by various methods on the Celeba-HQ, Places2, and our custom Thangka dataset, respectively.

In the comparison against existing state-of-the-art methods on the Celeba dataset, our model demonstrates notable differences. As seen in Fig. 6, methods like EC and CTSDG,

Table 1 Settings of Celeba, Facade and Thangka datasets

Dataset	Training set	Validation set	Test set
Celeba-HQ	27,000	1800	1200
Places2	90,000	6000	4000
Thangka	2500	300	200

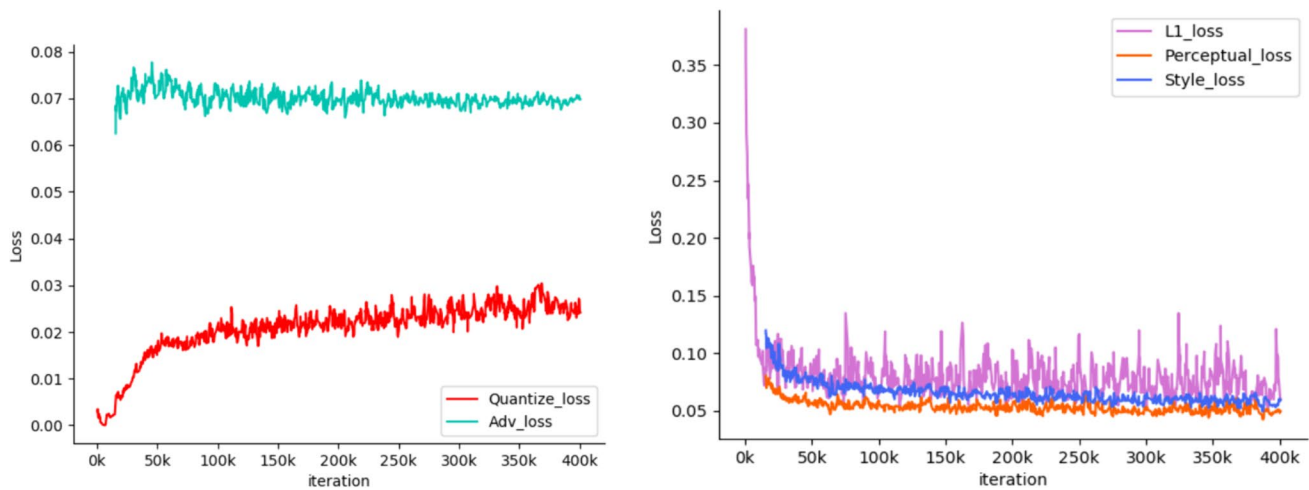


Fig. 5 Curve of loss values over iteration counts

when confronted with images featuring extensive damage, inadequately predict structures, leading to significant distortions in the restored outcomes. For instance, in rows 3 and 5 of Fig. 6b and c, noticeable omissions are present in the cheeks and eyes of the subjects. ICT, which leverages a transformer for visual prior reconstruction, produces structurally sound restorations overall but lacks refinement in detail. In Fig. 6d, deformities in the restored frames (first and second rows) and asymmetry in the eyes are evident. MAT, a mask-guided transformer model for large-area damage repair, does not perform satisfactorily in handling smaller missing regions within images. In Fig. 6e, the restored hair and eyes in rows 4 and 5 do not adhere well to facial characteristics. PUT's P-VQVAE encoder, which transforms images at original resolution into latent features in a non-overlapping manner to avoid cross-information influence, fails to fully comprehend semantic features. When restoring the bangs of the character in the second image of Fig. 6, our method not only successfully recovers the side bangs but also ensures a reasonable extent of the bangs, with hair that is finely textured and does not obscure the eyes. In contrast, the PUT method produces disheveled bangs that partially cover the eye, showing a clear discrepancy from the original image and disrupting the overall harmony of the picture. In the fourth image of Fig. 6, we tested the background restoration after hat removal. Although the PUT method seemingly restores the shape of the hat, the texture and color of the hat are incongruous, resulting in a poorer overall visual effect. Our CDCT method, leveraging the content of the known areas of the image, successfully reconstructs the person's hair, achieving a good balance between realism and fidelity of details, aligning well with human visual perception. In contrast, our proposed algorithm, by incorporating the concept of vector quantization alongside the parallel CSWin transformer module and multi-scale feature guidance

module, demonstrates restorations with sharp edges and naturally transitioning colors. Even in severely damaged areas, the semantic content of the repairs is reasonable, avoiding any discordant or abrupt local inconsistencies.

Figure 7 illustrates the inpainting effects of various models on the places2 dataset. EC and CTSDG, unable to capture long-range features, produce blurry and inconsistently bordered artifacts. ICT, due to substantial information loss during downsampling, results in defective horse legs being restored. MAT and PUT's repair outcomes exhibit semantic inconsistencies and color discrepancies; for instance, as shown in Fig. 7e, a cabinet is generated on the grassland in the third row, which is semantically inappropriate. In Fig. 7f, the fourth row depicts an unnatural restoration of the background after removing a person, where the generated rocks appear unrealistic. Our proposed method, through shared codebook learning, avoids image information loss, thereby acquiring richer semantic information and enabling high-fidelity image inpainting.

Figure 8 and 9 illustrates the repair comparison for various damaged regions of thangkas. The EC algorithm, when confronted with large areas of damage, is limited by its small receptive field, leading to the repair results with extensive texture blurring and an inability to reconstruct image structures. As shown in Fig. 9b, EC restoration fails, exhibiting semantic loss phenomena such as missing petals and green leaves. The CTSDG algorithm, in dealing with local missing regions of figures, can leverage edge information to reconstruct the basic outline of the figures. However, it falls short of restoring material characteristics and fine details at a micro level. From Fig. 9c, it is observed that the ICT method's restoration result exhibits noticeable artificial restoration artifacts, failing to reproduce the fine textures and color gradients present in the original image. As shown in Fig. 8e, the MAT algorithm demonstrates strong restoration

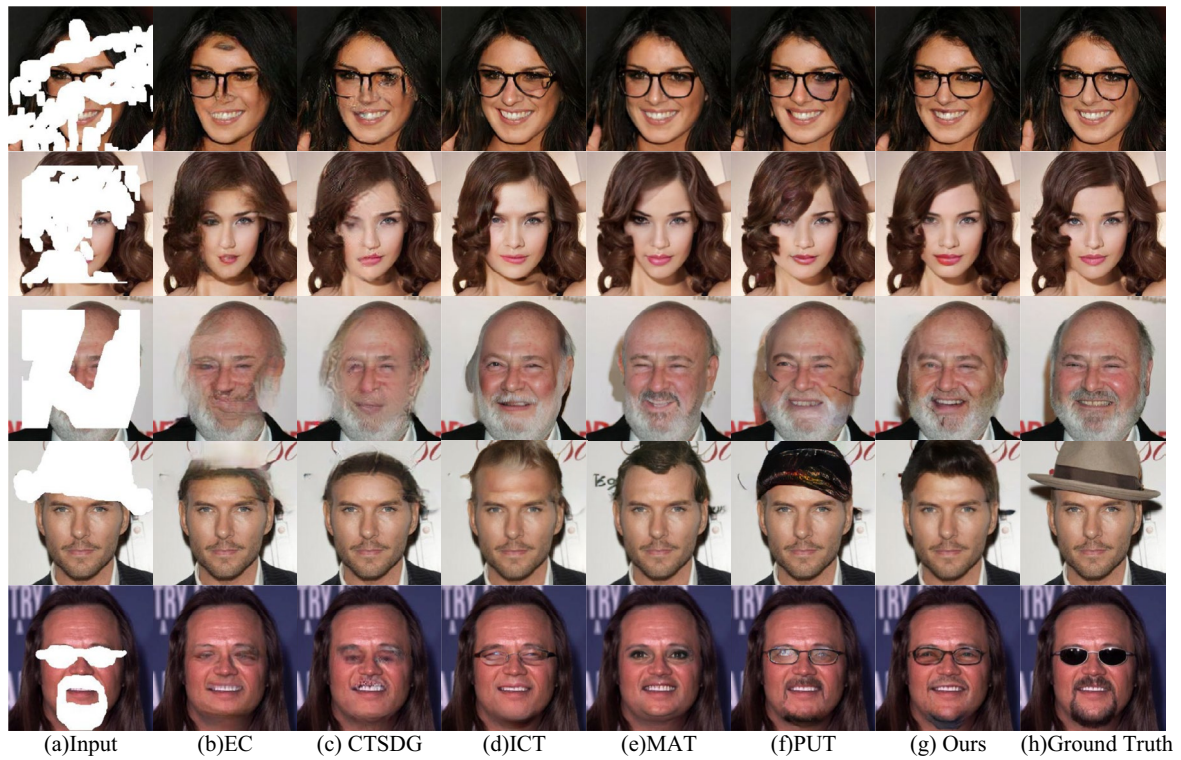


Fig. 6 Qualitative comparison results on the Celeba-HQ dataset

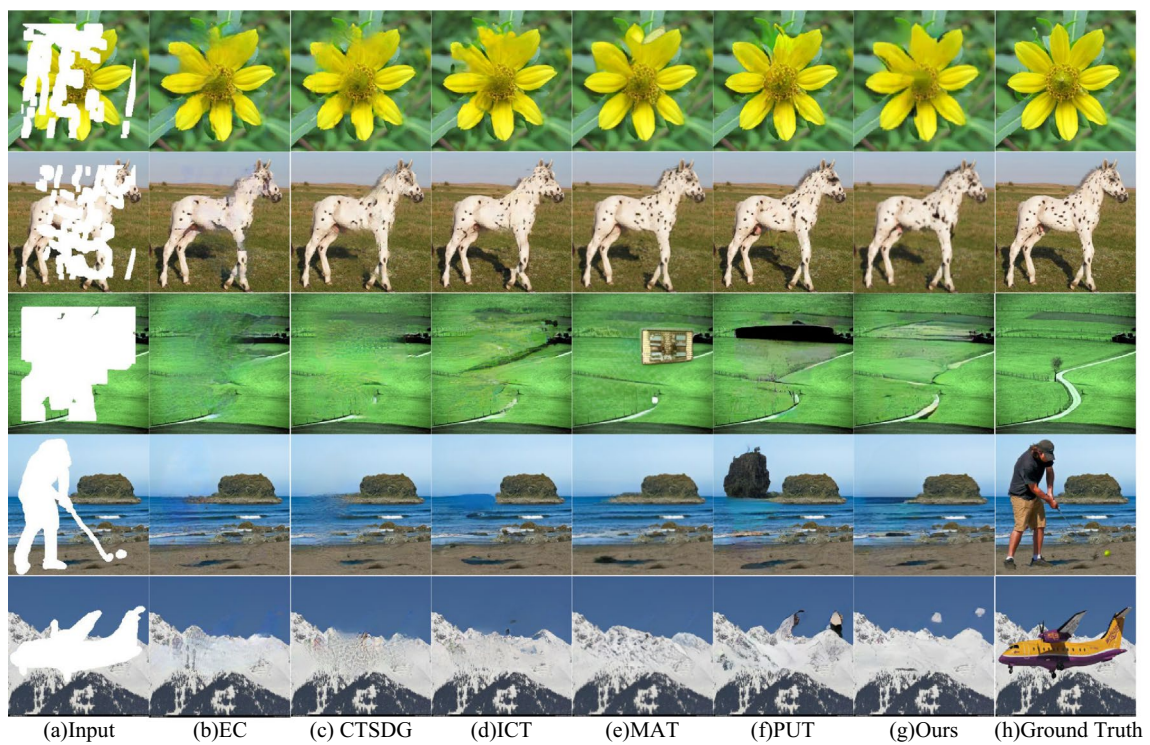


Fig. 7 Qualitative comparison results on the places2 dataset

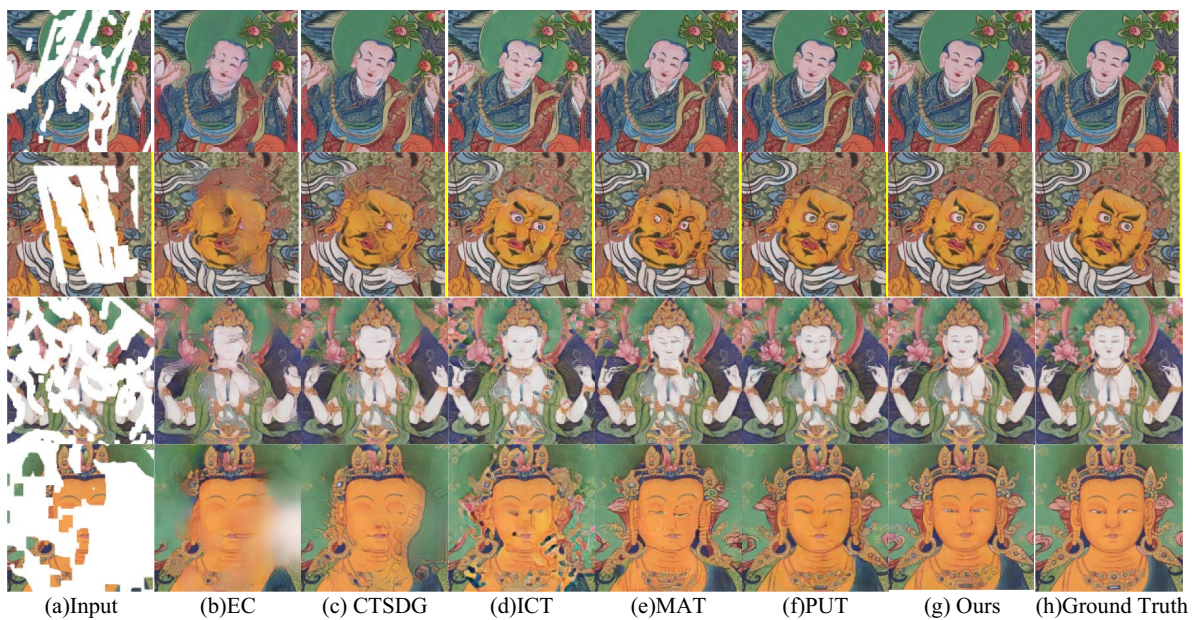


Fig. 8 Qualitative comparison results for character images in the Thangka dataset

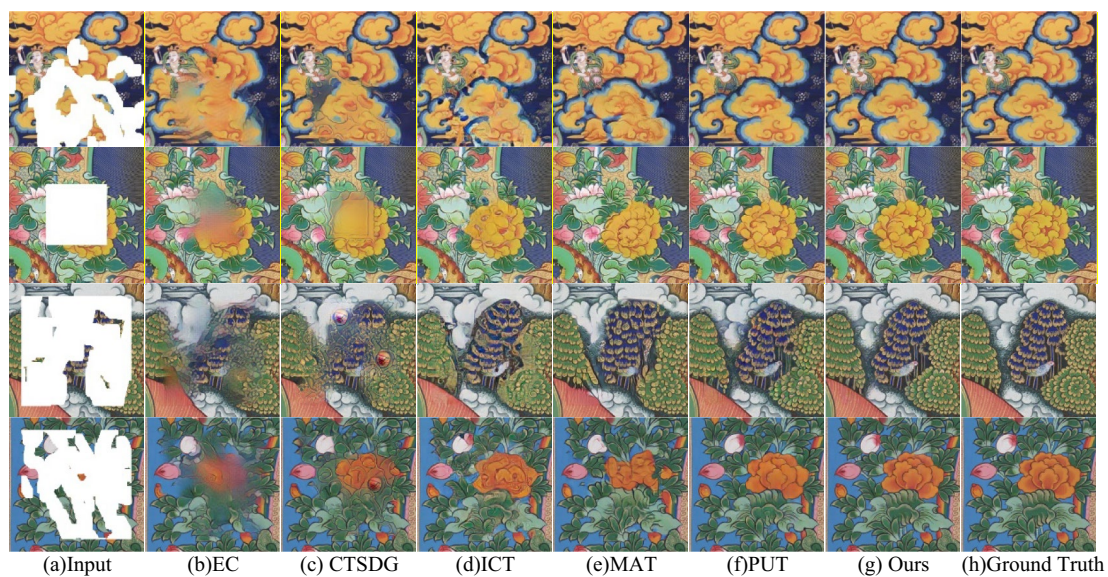


Fig. 9 Qualitative comparison results for non-character images in the Thangka dataset

capabilities even when the area of damage is substantial, repairing outcomes that the first two algorithms failed to restore in the second row and fourth row's eye regions. Nevertheless, the positioning of the eyes is still unreasonable, and there is distortion in the face. As can be seen from the second and fourth rows of Fig. 8, our method, by incorporating the Parallel CSWin Transformer module, is able to accurately reconstruct the contours of the eyes and the shape of the lips, ensuring clear lip edges with distinct gradations of color, closely resembling the artistic style of the original

Thangka. In contrast, the eye socket structure restored by PUT is blurred, and the color transition in the mouth area is unnatural. Visually, after restoration using the algorithm proposed in this paper, all five images exhibit consistency in structural continuity and precision in texture details, aligning closely with the original images. Consequently, it is verified that the algorithm presented herein is better suited for the restoration of thangka paintings, which are characterized by complex textures and rich colors.

4.3 Quantitative comparisons

Given that individual perceptions and evaluation criteria may vary, quantitative comparisons provide a more accurate reflection of subtle differences in performance, enhancing the verifiability and reproducibility of research findings. We have adopted four assessment standards—PSNR, SSIM, MAE, and LPIPS—and conducted experiments on datasets including Celeba-HQ, Places2, and a custom thangka dataset. Among them, PSNR (Peak Signal-to-Noise Ratio) is a commonly used objective measurement method for image quality, such as for measuring the difference between the restored image and the original image. A higher PSNR value indicates better restored image quality. However, the assessment results may differ from human perception. It is specifically defined as shown in Eq. (8).

$$\text{PSNR} = 10 * \log_{10} \left(\frac{\text{MaxValue}^2}{\text{MSE}} \right) \tag{8}$$

where, MSE represents the mean squared error between two images; MaxValue is the maximum value that a pixel in the image can attain. SSIM (Structural Similarity) evaluates the similarity between images using three factors that conform to human perception, namely luminance, contrast, and structure. This method is more consistent with human visual perception than other approaches, albeit with a higher computational cost. The SSIM calculation formula is as follows in Eq. (12):

$$l(X, Y) = \frac{2\mu_X\mu_Y + C_1}{\mu_X^2 + \mu_Y^2 + C_1} \tag{9}$$

$$c(X, Y) = \frac{2\sigma_X\sigma_Y + C_2}{\sigma_X^2 + \sigma_Y^2 + C_2} \tag{10}$$

$$s(X, Y) = \frac{\sigma_{XY} + C_3}{\sigma_X\sigma_Y + C_3} \tag{11}$$

$$\text{SSIM}(X, Y) = l(X, Y) * c(X, Y) * s(X, Y) \tag{12}$$

where, the mean of X is denoted by μ_X , and its standard deviation by σ_X , with Y following the same convention. The covariance of X and Y is also denoted by σ_{XY} . Formula (6) represents the luminance factor, formula (7) represents the contrast factor, and formula (8) represents the structural factor. Constants C_1 , C_2 , and C_3 are used to avoid division by zero. LPIPS (Learned Perceptual Image Patch Similarity) is a state-of-the-art perceptual metric based on human judgments of similarity. To compute the LPIPS metric, features are first extracted using a given network and normalized per channel. Activations are scaled by vector w_l sequentially

across channels, and the l_2 distance is computed. Then, we average over spatial dimensions and all layers. A higher LPIPS score indicates greater dissimilarity between the generated image and the original one, whereas a lower score indicates greater similarity. LPIPS is specifically represented by formula (13).

$$d(I, I_0) = \sum_l \frac{1}{H_l W_l} \sum_{h,w} \left\| w_l \odot (\hat{Y}_{hw}^l - \hat{Y}_{0hw}^l) \right\|_2^2 \tag{13}$$

where, I denotes the original image patch, and I_0 denotes the generated image patch. \odot represents the Hadamard product, which is used for element-wise multiplication of matrices. $\hat{Y}^l, \hat{Y}_0^l \in \mathbb{R}^{H_l \times W_l \times C_l}$ represent the unit-normalized outputs of I and I_0 at layer l , respectively. The vector $w_l \in \mathbb{R}^{C_l}$ is a trainable weight parameter, and when $w_l = 1 \forall l$, it is equivalent to computing the cosine distance. MAE (Mean Absolute Error) refers to the average of the absolute differences between two values, which prevents positive and negative errors from cancelling each other out. A smaller value indicates higher model accuracy, and it is specifically defined as shown in Eq. (14).

$$\text{MAE} = \frac{1}{m} \sum_{i=1}^m |Y_i - \hat{Y}_i| \tag{14}$$

where, m denotes the number of pixels, Y_i represents the actual values, and \hat{Y}_i stands for the predicted values.

All test images were uniformly set to a resolution of 256×256 , and they were subjected to irregular masks of the same proportion. We compared existing mainstream algorithms such as EC, CTSDG, ICT, MAT, and PUT, with the algorithm proposed in this paper. Based on these comparisons, we compiled specific numerical values for each evaluation metric, as shown in Table 2.

Analysis of the results in Table 2 reveals that, in both the Places2 scene dataset and our custom thangka dataset, our algorithm outperforms others, demonstrating a significant advantage in terms of similarity at both pixel level and structural level. In some instances, discrepancies between objective evaluation metrics and intuitive visual observations are noted, which underscores the limitations of relying solely on either objective or subjective measures to gauge the quality of image inpainting. At the same time, it also strongly proves the rationality and necessity of combining the two evaluation methods for comprehensive assessment in this paper.

4.4 Ablation study

To validate the effectiveness of the key components in the proposed method, a series of ablation experiments were conducted on our custom thangka dataset. The main experiments include the following: (b) The encoder part of the

Table 2 Objective quantitative comparison of this paper's algorithm with EC, CTSDG, ICT, MAT, PUT on three datasets with different mask ratios

Method	Dataset	PSNR↑			SSIM↑			LPIPS↓			MAE↓		
		20–40%	40–60%	Random	20–40%	40–60%	Random	20–40%	40–60%	Random	20–40%	40–60%	Random
EC	Celeba-HQ	26.611	21.875	21.504	0.928	0.818	0.799	0.0685	0.1420	0.1448	0.0216	0.0481	0.0366
CTSDG		28.383	24.244	23.216	0.951	0.874	0.845	0.0609	0.1401	0.1354	<i>0.0165</i>	0.0353	0.0285
ICT		27.781	23.265	22.742	0.948	0.851	0.864	0.0513	0.1087	0.1066	0.0212	0.0425	0.0319
MAT		29.737	25.158	<i>24.657</i>	0.962	0.906	0.890	0.0525	0.0933	0.1021	0.0174	0.0335	0.0259
PUT		28.232	23.603	24.599	0.954	0.871	0.898	0.0415	0.1028	<i>0.0940</i>	0.0176	0.0384	<i>0.0254</i>
Ours		<i>30.316</i>	<i>25.807</i>	<i>24.056</i>	<i>0.964</i>	<i>0.911</i>	<i>0.900</i>	<i>0.0368</i>	<i>0.0708</i>	0.0968	0.0179	<i>0.0324</i>	0.0289
EC	Places2	23.248	19.236	20.778	0.843	0.661	0.742	0.1317	0.2474	0.1792	0.0310	0.0636	0.0467
CTSDG		19.262	19.055	<i>22.206</i>	0.798	0.687	<i>0.786</i>	0.1734	0.2430	0.1922	0.0546	0.0628	<i>0.0388</i>
ICT		22.794	18.257	20.097	0.834	0.624	0.725	0.1170	0.2247	0.1729	0.0357	0.0737	0.0537
MAT		23.982	18.882	21.421	0.861	0.672	0.770	0.1485	0.2553	0.1955	0.0326	0.0690	0.0480
PUT		22.830	17.953	20.914	0.843	0.653	0.783	<i>0.0808</i>	0.2114	<i>0.1419</i>	0.0276	0.0666	0.0426
Ours		23.971	<i>19.582</i>	20.605	<i>0.872</i>	<i>0.742</i>	0.765	0.0852	<i>0.1653</i>	0.1457	<i>0.0270</i>	<i>0.0471</i>	0.0456
EC	Thangka	22.880	20.615	20.378	0.780	0.695	0.683	0.1387	0.2218	0.2241	0.0349	0.0454	0.0505
CTSDG		24.180	21.874	21.615	0.832	0.767	0.754	0.1076	0.1777	0.1879	0.0282	0.0366	0.0420
ICT		23.787	22.073	21.171	0.851	0.812	0.799	0.0956	0.1333	0.1497	0.0393	0.0430	0.0495
MAT		23.111	21.197	20.731	0.806	0.749	0.732	0.0933	0.1293	0.1506	0.0316	0.0389	0.0468
PUT		27.858	26.460	25.576	0.928	0.919	0.905	0.0402	0.0467	0.0635	0.0194	0.0222	0.0267
Ours		<i>30.064</i>	<i>29.260</i>	<i>27.811</i>	<i>0.962</i>	<i>0.956</i>	<i>0.947</i>	<i>0.0209</i>	<i>0.0246</i>	<i>0.0352</i>	<i>0.0181</i>	<i>0.0189</i>	<i>0.0226</i>

The best results are marked in italics, followed by those marked in bold

CDCT model in this paper uses the Conv layer of the same size to replace the Linear layer, (c) The parallel CSWin transformer module is replaced by the same number of standard transformer modules, (d) The parallel structure of the standard self-attention and CSWin attention in the PCT module is changed to a serial structure, (e) The multi-scale feature guidance module is removed, (f) The LKA structure in the multi-scale feature guidance module is replaced by the Conv layer, and (g) The complete network structure of this paper.

Table 3 shows the results of the objective evaluation of the different component ablation studies. Variants 1 and 2 use encoders sourced from VQGAN as well as the standard transformer module, which allows for over-compression of information as well as under-utilization of local details, which affects the performance of the model. Variant 3 is a

change from a parallel to a sequential approach to the internal structure of the PCT module, with a small decrease in metrics. Variants 4 and 5 demonstrate that the multi-scale feature guidance module maintains the ability to decode latent representations while making full use of non-masked region features. Relative to the other replacement components, the complete model with the addition of the linear residual encoder module, the parallel CSWin transformer module, and the multi-scale feature guidance module show an average improvement of 1.741 dB and 0.038 in the PSNR and SSIM values, and an average decrease of 0.0221 and 0.0053 in the LPIPS value as well as the MAE value. This indicates that these improved modules have a positive impact on the quality of the repair results.

Figure 10 showcases the visual outcomes of various components in our model. As depicted in Fig. 10b, Variant

Table 3 Quantitative ablation analysis of the proposed method on the custom thangka dataset

Types	Components	PSNR/dB↑	SSIM↑	LPIPS↓	MAE↓
Variant 1	+ Convolutional Encoder	25.792	0.901	0.0583	0.0335
Variant 2	+ Standard Transformer	25.938	0.904	0.0560	0.0254
Variant 3	+ Serial CSWin Transformer Module	27.318	0.924	0.0403	0.0225
Variant 4	– Multi-scale Feature Guidance Module	21.453	0.781	0.1227	0.0643
Variant 5	+ Convolutional Guidance Module	26.650	0.913	0.0509	0.0234
Our	Proposed Complete Architecture	28.165	0.948	0.0292	0.0209

Bold markings signify the optimal performance on the assessment

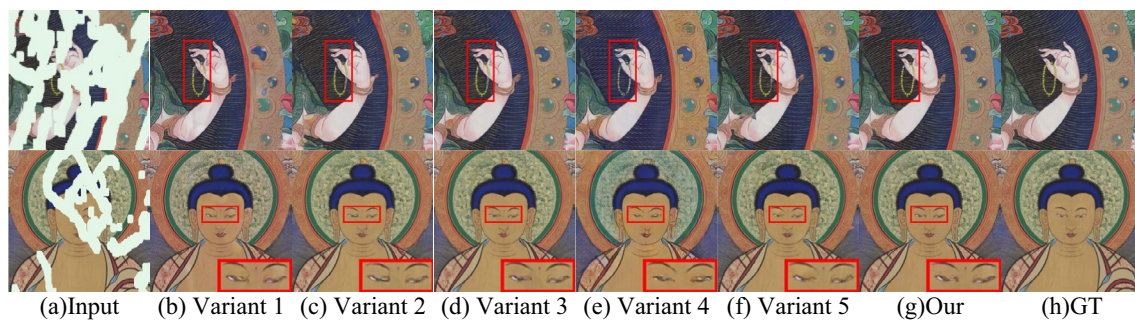


Fig. 10 Analysis of visual effects for individual components of the CDCT model (color figure online)

1 exhibits a lack of coherence between the damaged area and its surroundings, with noticeable variations in brightness and skin tone across the subject’s arms, face, and chest. Both Variant 2 and Variant 3 reveal inconsistencies in the size of beads held by the figure, introducing artifacts into the restoration. As seen in Fig. 10e, the absence of the multi-scale feature guidance module leads to a reduction in local informative details, with the figure’s fingers being influenced by the surrounding blue background, and the edges of the eyes appearing distorted with unnatural transitions. Figure 10g validates the effectiveness and superiority of our proposed CDCT algorithm in tackling images with complex color schemes, demonstrating that it yields more realistic and logically coherent repair outcomes.

In the first-stage network, our proposed CDCT model embeds continuous features into a discrete space of finite size, comprised of k -code vectors. This section conducts an ablation study to investigate the impact of the number of code vectors (k) in the codebook on model performance. Table 4 illustrates that a codebook size of 1024 on the thangka dataset yields better results, proving more effective in enhancing reconstruction quality rather than suggesting that larger codebook vectors inherently lead to more reasonable data compression.

This study undertook five experimental groups to ascertain the optimal hyperparameters for the number of attention heads and the embedding dimension. It was found that setting the number of attention heads in the PCT module to 8 and the embedding dimension to 512 enabled the model to more effectively capture long-range dependencies in the input sequences, significantly improving across all four evaluation metrics, while also avoiding an excessively high embedding dimension that would increase the computational burden of the model, as shown in Table 5.

4.5 Analysis of computational overhead

To comprehensively evaluate the computational cost of our proposed method, this section conducts a comparative analysis of the CDCT method against others in terms of parameter

Table 4 Impact of different codebook sizes on model performance

Codebook Sizes (k)	PSNR/dB \uparrow	SSIM \uparrow	LPIPS \downarrow	MAE \downarrow
512	26.033	0.839	0.0491	0.0250
1024	27.868	0.889	0.0311	0.0208
2048	26.889	0.868	0.0414	0.0216

Bold markings signify the optimal performance on the assessment

count, memory consumption, and inference time. We utilized the same experimental environment and datasets to ensure fairness in the comparison. In Table 6, “Total params” denotes the total number of parameters in the network models, including trainable and non-trainable parameters. “Total memory” indicates the amount of GPU memory space occupied during the testing phase of each method. “Runtime” refers to the time required by each method to restore a single image.

The results in Table 6 demonstrate that our CDCT method, when pitted against state-of-the-art methods employing Transformer architectures (such as ICT, MAT, PUT), not only exhibits a more streamlined parameter design but also significantly reduces memory footprint. Compared to MAT, our approach decreases the parameter count by 16% and slashes the GPU memory cost by 11%. When tackling the task of restoring a single image, our method also consumes less time.

5 Conclusion

This paper presents a novel image inpainting method that synergizes discrete codebooks with transformers, incorporating several distinctive design innovations. Firstly, a linear residual encoder replaces convolutional downsampling, enabling independent encoding of feature blocks to prevent cross-information interference. Unlike typical inpainting models, our approach employs a codebook to discretely encode intermediate model features. Secondly, to mitigate

Table 5 Performance comparison of different hyperparameter combinations of PCT module

Heads	Embedding dims	Params (M)	PSNR/dB↑	SSIM↑	LPIPS↓	MAE↓
4	512	53.05	26.684	0.892	0.0420	0.0218
8	512	53.05	27.752	0.908	0.0302	0.0200
16	512	53.05	26.158	0.874	0.0471	0.0231
8	256	20.95	23.487	0.822	0.0884	0.0292
8	768	106.13	26.174	0.875	0.0470	0.0230

Bold markings signify the optimal performance on the assessment

Table 6 Analysis of computational overhead for different methods

Method	Total params (M)↓	Total memory (MB)↓	Runtime (s)↓
EC	21.54	1745.32	2.142
CTSDG	52.14	3346.94	3.748
ICT	121.63	5558.47	28.72
MAT	60.35	4081.67	2.893
PUT	96.38	4124.00	13.57
Ours	50.61	3618.13	3.594

information loss in Transformers, the input is not the discrete tokens (indices), but rather the features outputted by the encoder, while the discrete tokens solely serve as the Transformer's output. Moreover, the design of the parallel CSwin transformer module enhances the accuracy of token prediction while reducing parameter count. An additional multi-scale feature guidance module is then incorporated into the decoder, effectively preserving local details in non-damaged areas and restoring details from the encoder's quantized output. Extensive experimentation across representative tasks validates the CDCT method's capability to handle color-diverse, semantically rich thangka images, as well as effectively repair various defects in natural images. Through meticulous ablation studies, the efficacy of the proposed model design is demonstrated.

While our method has achieved notable success in dealing with large occlusion regions and complex scenes, there remain certain limitations. Specifically, given our reliance on patch-level feature representations, the model is prone to being influenced by distracting information from the known background areas during the restoration process. For instance, when restoring the stamens of flowers in Fig. 7, the model can be interfered with by the features of adjacent petals, leading to overly smoothed restoration results lacking in detail. Similarly, when handling roads on grasslands in Fig. 7, due to the interference of grassland environmental features, the model struggles to precisely define and recover the boundaries of the road.

Author contributions JB: Literature search, experimentation, data analysis, and drafting of the manuscript. YF: Conceptualization of the study, design of methods, overall supervision, rigorous review, and editing. ZZ: Study design, data acquisition, and modification of methods. All authors reviewed and approved the final manuscript.

Funding This work was funded by the National Natural Science Foundation of China (No. 62062061, No. 62263028) and the Xizang Minzu University Internal Research Project (No. 324132400307).

Data availability For publicly accessible data resources, our study employed the [Celeba-HQ dataset] and the [Places2 dataset]; further details can be found in references [48] and [49], respectively. Moreover, the [Thangka dataset] used in this research was independently constructed by the authors. Due to the sensitivity of the data and considerations for the continuity of upcoming research projects, this self-made Thangka dataset is presently not available for public access. We recognize the importance of data sharing in facilitating scientific validation and progress; hence, we are willing to discuss the potential for data sharing with scholars who have particular research requirements.

Declarations

Conflict of interest The authors declare no competing interests.

Ethical approval Not applicable, as the research did not involve human subjects, animals, or the collection of personal data.

References

- Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, pp. 417–424 (2000)
- Chan, T.F., Shen, J.: Nontexture inpainting by curvature-driven diffusions. *J. Vis. Commun. Image Represent.* **12**(4), 436–449 (2001)
- Komodakis, N., Tziritas, G.: Image completion using efficient belief propagation via priority scheduling and dynamic pruning. *IEEE Trans. Image Process.* **16**(11), 2649–2661 (2007)
- Cui, Y., Ren, W., Cao, X., Knoll, A.: Image restoration via frequency selection. *IEEE Trans. Pattern Anal. Mach. Intell.* **46**(2), 1093–1108 (2023)
- Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A. A.: Context encoders: feature learning by inpainting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2536–2544 (2016)

6. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Globally and locally consistent image completion. *ACM Trans. Graph.* **36**(4), 1–14 (2017)
7. Wang, Y., Tao, X., Qi, X., Shen, X., Jia, J.: Image inpainting via generative multi-column convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **31** (2018)
8. Cui, Y., Ren, W., Yang, S., Cao, X., Knoll, A.: Irnext: rethinking convolutional network design for image restoration. In: *International Conference On Machine Learning* (2023)
9. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T. S.: Generative image inpainting with contextual attention. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5505–5514 (2018)
10. Sagong, M. C., Shin, Y. G., Kim, S. W., Park, S., Ko, S. J.: Pepsi: Fast image inpainting with parallel decoding network. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11360–11368 (2019)
11. Xiong, W., Yu, J., Lin, Z., Yang, J., Lu, X., Barnes, C., Luo, J.: Foreground-aware image inpainting. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5840–5848 (2019)
12. Guo, X., Yang, H., Huang, D.: Image inpainting via conditional texture and structure dual generation. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14134–14143 (2021)
13. Dong, Q., Cao, C., Fu, Y.: Incremental transformer structure enhanced image inpainting with masking positional encoding. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11358–11368 (2022)
14. Bai, J., Fan, Y., Zhao, Z., Zheng, L.: Image inpainting technique incorporating edge prior and attention mechanism. *Comput. Mater. Contin.* **78**(1) (2024)
15. Song, Y., Yang, C., Shen, Y., Wang, P., Huang, Q., Kuo, C. C. J.: Spg-net: segmentation prediction and guidance network for image inpainting. *arXiv preprint arXiv:1805.03356* (2018)
16. Liao, L., Xiao, J., Wang, Z., Lin, C. W., Satoh, S.I.: Guidance and evaluation: semantic-aware image inpainting for mixed scenes. In: *Computer Vision—ECCV 2020: 16th European Conference*, pp. 683–700. Springer, Berlin (2020)
17. Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., Sutskever, I.: Generative pretraining from pixels. In: *International Conference on Machine Learning*, pp. 1691–1703 (2020)
18. Chen, Y., Dai, X., Chen, D., Liu, M., Dong, X., Yuan, L., Liu, Z.: Mobile-former: Bridging mobilenet and transformer. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5270–5279 (2022)
19. Yu, Z., Li, X., Sun, L., Zhu, J., Lin, J.: A composite transformer-based multi-stage defect detection architecture for sewer pipes. *Computers, Materials & Continua* **78**(1) (2024)
20. Peng, Y., Zhang, Y., Xiong, Z., Sun, X., Wu, F.: GET: group event transformer for event-based vision. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6038–6048 (2023)
21. Wan, Z., Zhang, J., Chen, D., Liao, J.: High-fidelity pluralistic image completion with transformers. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4692–4701 (2021)
22. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30**, 5998–6008 (2017)
23. Esser, P., Rombach, R., Ommer, B.: Taming transformers for high-resolution image synthesis. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12873–12883 (2021)
24. Zhu, Z., Feng, X., Chen, D., Bao, J., Wang, L., Chen, Y.: Designing a better asymmetric vqgan for stablediffusion. *arXiv preprint arXiv:2306.04632* (2023)
25. Zheng, C., Vuong, T.L., Cai, J., Phung, D.: Movq: Modulating quantized vectors for high-fidelity image generation. *Adv. Neural Inf. Process. Syst.* **35**, 23412–23425 (2022)
26. Yoo, Y., Choi, J.: Topic-VQ-VAE: leveraging latent codebooks for flexible topic-guided document generation. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38, pp. 19422–19430 (2024)
27. Liu, Q., Tan, Z., Chen, D., Chu, Q., Dai, X., Chen, Y.: Reduce information loss in transformers for pluralistic image inpainting. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11347–11357 (2022)
28. Zheng, C., Song, G., Cham, T. J., Cai, J., Phung, D., Luo, L.: High-quality pluralistic image completion via code shared vqgan. *arXiv preprint arXiv:2204.01931* (2022)
29. Van Den Oord, A., Vinyals, O.: Neural discrete representation learning. *Adv. Neural Inf. Process. Syst.* **30**, 6309–6318 (2017)
30. Cui, Y., Ren, W., Knoll, A.: Omni-kernel network for image restoration. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1426–1434 (2024)
31. Cui, Y., Tao, Y., Bing, Z., Ren, W., Gao, X., Cao, X., Knoll, A.: Selective frequency network for image restoration. In: *The Eleventh International Conference on Learning Representations* (2023)
32. Cui, Y., Ren, W., Cao, X., Knoll, A.: Focal network for image restoration. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13001–13011 (2023)
33. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D.: An image is worth 16x16 words: transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020)
34. Ho, J., Kalchbrenner, N., Weissenborn, D., Salimans, T.: Axial attention in multidimensional transformers. *arXiv preprint arXiv:1912.12180* (2019)
35. Dong, X., Bao, J., Chen, D., Zhang, W., Yu, N., Yuan, L.: Cswin transformer: a general vision transformer backbone with cross-shaped windows. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12124–12134 (2022)
36. Zheng, C., Cham, T. J., Cai, J., Phung, D.: 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 (2022)
37. Razavi, A., Van den Oord, A., Vinyals, O.: Generating diverse high-fidelity images with vq-vae-2. *Adv. Neural Inf. Process. Syst.* **32**, 14866–14876 (2019)
38. Chen, C., Shi, X., Qin, Y., Li, X., Han, X., Yang, T., Guo, S.: Real-world blind super-resolution via feature matching with implicit high-resolution priors. In: *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 1329–1338 (2022)
39. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: *Computer Vision—ECCV 2016: 14th European Conference*, pp. 694–711. Springer (2016)
40. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B.: Generative adversarial networks. *Commun. ACM* **63**(11), 139–144 (2020)
41. Gatys, L. A., Ecker, A. S., Bethge, M.: Image style transfer using convolutional neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2414–2423 (2016)
42. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014)
43. Bengio, Y., Léonard, N., Courville, A.: Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432* (2013)
44. Suvorov, R., Logacheva, E., Mashikhin, A., Remizova, A.: Resolution-robust large mask inpainting with Fourier convolutions. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2149–2159 (2022)

45. Guo, M.H., Lu, C.Z., Liu, Z.N., Cheng, M.M., Hu, S.M.: Visual attention network. *Computational Visual. Media* **9**(4), 733–752 (2023)
46. Zhang, B., Sennrich, R.: Root mean square layer normalization. *Adv. Neural Inf. Process. Syst.* **32**, 12360–12371 (2019)
47. Ramachandran, P., Zoph, B., Le, Q. V.: Searching for activation functions. arXiv preprint [arXiv:1710.05941](https://arxiv.org/abs/1710.05941) (2017)
48. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv preprint [arXiv:1710.10196](https://arxiv.org/abs/1710.10196) (2017)
49. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: a 10 million image database for scene recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**(6), 1452–1464 (2017)
50. Liu, G., Reda, F. A., Shih, K. J., Wang, T. C., Tao, A., Catanzaro, B.: Image inpainting for irregular holes using partial convolutions. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 85–100 (2018)
51. Nazeri, K., Ng, E., Joseph, T., Qureshi, F. Z., Ebrahimi, M.: Edge-connect: generative image inpainting with adversarial edge learning. arXiv preprint [arXiv:1901.00212](https://arxiv.org/abs/1901.00212) (2019)
52. Li, W., Lin, Z., Zhou, K., Qi, L., Wang, Y., Jia, J.: Mat: Mask-aware transformer for large hole image inpainting. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10758–10768 (2022)

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.