IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 23 August 2024, accepted 12 September 2024, date of publication 16 September 2024, date of current version 3 October 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3461782

TOPICAL REVIEW

Review of Deep Learning-Based Image Inpainting Techniques

JING YANG[1] **AND NUR INTAN RAIHANA RUHAIYEM**[2]
[1]Shanxi Datong University, Datong 037000, China
[2]School of Computer Sciences, Universiti Sains Malaysia, Gelugor 11800, Malaysia

Corresponding author: Nur Intan Raihana Ruhaiyem (intanraihana@usm.my)

This work was supported in part by the Fundamental Research Grant Scheme under Grant FRGS/1/2021/ICT04/USM/02/1/Ministry of Higher Education, in part by the Scientific and Technologial Innovation Programs of Higher Education Institutions in Shanxi (STIP) under Grant 2022: 441, and in part by Shanxi Datong University Basic Research Program (Youth) under Grant 2022: 30.

ABSTRACT The deep learning-based image inpainting models discussed in this review are critical image processing techniques for filling in missing or removed regions in static planar images, and they have been extensively researched and applied. However, due to the rapid advancements in deep learning technologies, existing review studies exhibit shortcomings: in 1) providing a comprehensive and systematic classification of strategies for these models; 2) analyzing loss functions and evaluation metrics; and 3) organizing and categorizing data and application domains. To address these three aspects, this paper provides a systematic literature review of deep learning-based inpainting models. First, we rigorously classify the strategies used in image inpainting networks into four categories: layer-based, connection-based, multi-network, and multi-modal inpainting strategies. This classification is determined by the composition of the models, ensuring it comprehensively covers the vast majority of inpainting network strategies. Second, to facilitate a more effective study and comparison of loss functions and evaluation metrics, we categorize them into pixel-based, feature-based, and model-based approaches, conducting a thorough analysis of each. Finally, we categorize image inpainting applications into three main types according to dataset characteristics: natural image inpainting, detection image inpainting, and artistic image inpainting, offering insights into each specific domain.

INDEX TERMS Image inpainting, deep learning.

I. INTRODUCTION

Images are an important medium for information dissemination. However, image data are susceptible to interference, noise, and damage, adversely affecting data analysis and knowledge extraction. Consequently, a range of image inpainting techniques has emerged to fill in missing parts, striving to generate high-quality images with continuity and reasonable content that meet the standards of human visual perception and scientific research. In the past two decades, image inpainting has developed into an important research area. It represents a key frontier in global research, holds significant academic value, and has been widely applied in fields such as remote sensing imagery, medical diagnosis, cultural heritage restoration, traffic image recovery, and artistic creation.

Image inpainting models discussed in this review primarily focus on filling in missing or damaged areas within 2D images, ensuring that the restored regions blend seamlessly with the surrounding content. These models deal with static, planar images and do not incorporate the temporal information found in frame sequences. From a practical perspective, the challenges in image inpainting can be divided into two categories. The first involves restoring damaged areas of an image to a complete state, such as in mural inpainting or old photograph inpainting. The second category focuses on removing unwanted regions from an image and filling them with plausible content, such as watermark removal or eliminating distractions in photo editing. For the model, both categories essentially involve filling in

The associate editor coordinating the review of this manuscript and approving it for publication was Hengyong Yu.

unnecessary regions with appropriate content. These tasks can be further classified based on the model's objective: one aims to produce a single and strictly accurate restoration result, while the other aims to generate multiple semantically plausible inpainting outcomes. Currently, based on algorithm, image inpainting algorithms can be divided into two main categories: traditional methods and those utilizing deep learning techniques. Traditional methods typically involve mathematical reasoning, using information from the original image to infer the missing data. Notable among these are the model proposed by Bertalmio, Sapiro, Casles, and Ballester (BSCB) [1] and the Criminisi model [2]. These models have inspired a series of similar models, such as the Total Variation (TV) model [3] and the PatchMatch model [4]. Although traditional image inpainting algorithms have achieved commendable success in correcting minor defects, they show limitations when dealing with extensive damage.

In recent years, driven by advancements in computer hardware and computational capabilities, deep learning-based image processing techniques have experienced significant growth. The advent of AlexNet [5], Visual Geometry Group (VGG) networks [6], ResNet [7], Generative Adversarial Networks (GAN) [8], and their improved models such as Wasserstein GAN (WGAN) [9], Wasserstein GAN-Gradient Penalty (WGAN-GP) [10], and Least Squares GAN (LSGAN) [11], has greatly advanced the development of image inpainting algorithms. Among these, early models like the Context Encoder [12], Globally and Locally Consistent Image Completion (GLCIC) [13] with global and local discriminators, Generative Multi-column Convolutional Neural Networks (GMCNN) [14] utilizing parallel multi-scale convolutions, Generative Image Inpainting with Contextual Attention (CA) [15] model incorporating self-attention mechanisms, and the Edge Connect model [16] with edge-assisted inpainting, all employ encoder-decoder structures. These models also incorporate GAN's discriminator to optimize the Generative Adversarial Loss between inpainted and real images. Consequently, in early review literature, these models were classified either as encoder-decoder type or GAN-based inpainting models. Additionally, some image inpainting models adopt the U-Net [17] architecture, such as the Shift-Net [18], the Deep Fusion Network (DFNet) [19] with fusion modules, the Partial Convolutions [20] designed for random defects, and the Pyramid-Context Encoder Network (PEN-Net) [21] employing pyramid structure. These models are categorized under the U-Net class. Many models within this class also utilize GAN's generative adversarial loss to constrain the generated results, making the restored images more realistic and credible. Hence, in many prior studies, these models were also classified as GAN-based image inpainting models. Subsequently, inspired by the exceptional performance of the Transformer model [22] in natural language processing, some researchers applied this attention-based architecture to image inpainting, resulting in a series of large-scale image inpainting models, including the TransFill model [23], MAE model [24], and ZITS model [25]. These models are generally classified as Transformer-based image inpainting models. Another class of models based on Denoising Diffusion, such as the Denoising Diffusion Probabilistic Model (DDPM) [26] and the Denoising Diffusion Implicit Models (DDIM) [27], are often not included in many review studies. Therefore, based on a review of a series of deep learning-based image inpainting models, we find that previous research and classification of image inpainting models are not sufficiently scientific and rigorous, with specific deficiencies as follows:

(1) Shortcomings in Systematic and Comprehensive Classification. Many prior studies categorize deep learning-based models into four major groups: Encoder-Decoder, U-Net, GAN, and Transformer. However, the classification of Encoder-Decoder and U-Net models is based on network architecture, while GANs are categorized by their loss functions, and Transformers by the type of network layers. This inconsistency in classification implies that most models can be classified into multiple categories, which hinders researchers from accurately understanding the characteristics and performance of these models. Furthermore, given the rapid advancement of deep learning image algorithms, a more comprehensive survey that encompasses a broader range of studies in deep learning image analysis is urgently needed.

(2) Insufficient Systematic Research on Loss Functions and Evaluation Metrics. Previous reviews have mainly listed the formulas and functions of loss functions and evaluation metrics without detailed analysis and organization. In fact, evaluation metrics and loss functions are closely related and can be studied together for comparative research. This approach can help select more effective loss functions based on evaluation metrics or design more practically valuable evaluation metrics based on loss functions.

(3) Insufficient Analysis of Data and Applications in Image Inpainting. The datasets used for image inpainting learning and training are closely related to their application domains. However, most previous reviews have only listed datasets and specific application areas without providing a detailed synthesis and analysis of their interrelation. Such an integrated summary would assist researchers in identifying suitable image inpainting models and strategies for their specific fields.

Based on the above reasons, this paper aims to provide a more comprehensive and systematic classification of deep learning-based image inpainting models by categorizing the strategies used by these models. These strategies are divided into those applied to the generator and those applied to the loss functions and evaluation metrics.

Firstly, addressing issues (1), we explore the generator models. The basic building blocks of a generator are layers and connections, so we analyze strategies applied to these two aspects. On the basis of networks composed of layers and connections, we identify that some generators are composed

of multiple networks, thus we classify strategies related to network composition as a third category. Regarding generator functionality, most studies indicate that generators typically produce a single inpainting result, whereas multiple inpainting outcomes are more valuable in many practical applications. Therefore, the fourth category addresses multi-output inpainting models.

Secondly, concerning issue (2), we examine loss functions and evaluation metrics. These can be broadly categorized into three types: pixel-based, feature-based, and model-based types. Pixel-based loss functions and metrics focus on pixel-level similarity. Feature-based loss function and metrics assess whether the inpainted and target images share similar features in pre-trained models, or other features similar to frequency characteristic. Model-based loss function, inspired by GAN principles, use learnable neural networks to judge the similarity between inpainted and target images, and model-based metrics evaluate the effectiveness of the restored image by training a complete deep learning network.

Finally, concerning issue (3) we delve into datasets and application domains. We analyze and classify the applications of deep learning-based image inpainting models according to the characteristics of datasets. Applications are divided into three main categories: natural images, detection images, and artistic images. Natural images include those that resemble what human eyes see, such as portraits, street scenes, landscapes, and aerial or satellite images. Detection images refer to non-natural images with research information obtained through detection instruments or secondary processing, such as medical images, waveforms, and heatmaps. Artistic images encompass modern and ancient artwork, including irreplaceable ancient artifacts like murals. Lastly, we analyze the current limitations of deep learning-based image inpainting methods and provide prospects for future development.

Based on the above, our contributions are summarized as follows:

Contribution 1: We provide a more rigorous and reasonable classification of strategies used in deep learning-based image inpainting models.

Contribution 2: We systematically categorize loss functions and evaluation metrics and conduct the first comparative study between them.

Contribution 3: We link datasets with image inpainting application domains and categorize them by image type, offering more targeted references for future research.

II. GENERATOR

The generator is a crucial component of image inpainting models, primarily composed of a series of layers and connections arranged in a specific order. In other words, the performance of an image inpainting model's generator is determined by three main factors: layers, connections, and their arrangement. Typically, the arrangement within a single network shows minimal variation, but numerous studies have discovered that employing multiple networks can significantly enhance model performance. Therefore, this chapter systematically classifies and reviews existing model strategies based on these three key factors: layers, connections, and multi-network configurations. Additionally, from the perspective of inpainting objectives, most studies design models for generating a single inpainting result. However, in certain scenarios, diversified inpainting results are more beneficial for research, necessitating models capable of producing multiple inpainting outcomes. To achieve diversified inpainting results, specific strategies are employed to help the model sample multiple random values from the likelihood distribution, thus generating varied but plausible outcomes. Therefore, this paper categorizes these strategies as the fourth type. The detailed classification of generator strategies is illustrated in the Figure 1. Classification figure of strategies used in the generator:

A. LAYERS

The primary function of layers in neural networks is to progressively transform input data into higher-level feature representations through a series of weight adjustments, biases, and nonlinear activation functions. This hierarchical feature extraction enables neural networks to perform complex tasks. Each layer extracts different features from the data and passes these features to subsequent layers for further processing. In image inpainting networks, neuron layers are mainly composed of convolutional layers and attention-based layers. In practical image inpainting models, these two types of layers can also be used in combination. In recent years, many studies have opted to use attention layers for learning structures and convolutional layers for learning textures.

1) CNN LAYERS

Convolutional Neural Networks (CNNs) are highly effective in image tasks due to their ability to extract local features using convolution operations with a sliding window. By leveraging parameter sharing, CNNs reduce the number of model parameters, facilitating the training of smaller models that perform well even on smaller datasets. Supported by a series of classic convolutional model architectures such as VGG [6], ResNet [7], and a series of Inception networks [28], [29], [30], many early image inpainting models are based on CNNs. Typically, CNNs are composed of multiple blocks, each containing convolutional layers, normalization layers, and activation function layers. Given that most research focuses on improving convolutional layers, this paper primarily discusses strategies applied to the convolutional layers. In this section, the convolution algorithms are categorized into Dilated Convolution, Partial Convolution, Gated Convolution, Region-wise Convolution, and Hypergraph Convolution, based on the study of image inpainting models and the strategies most commonly used for convolution.

a: DILATED CONVOLUTION

In deep learning networks, pooling operations are typically employed to increase the receptive field with normal

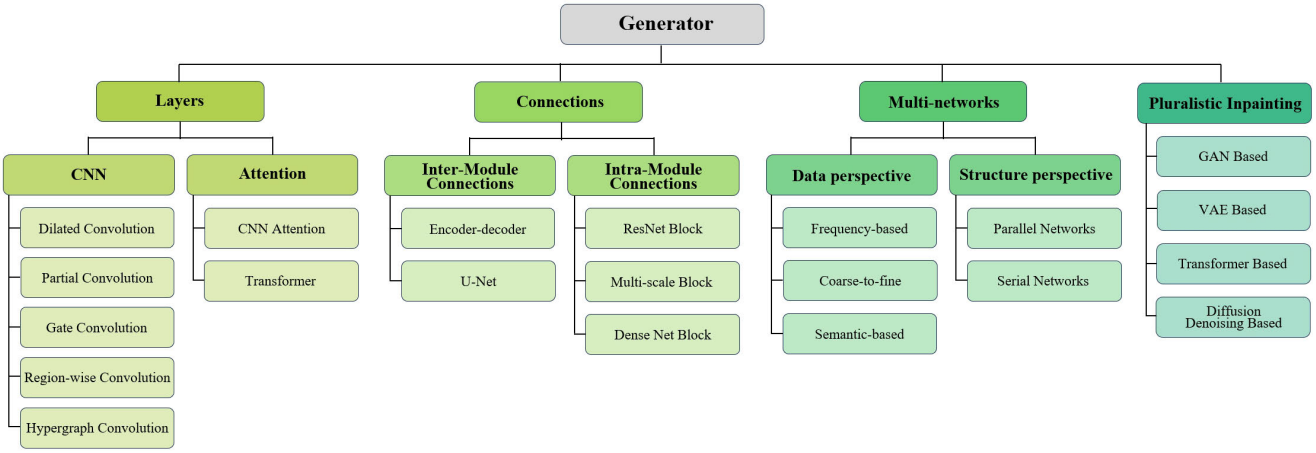

FIGURE 1. Classification figure of strategies used in the generator.

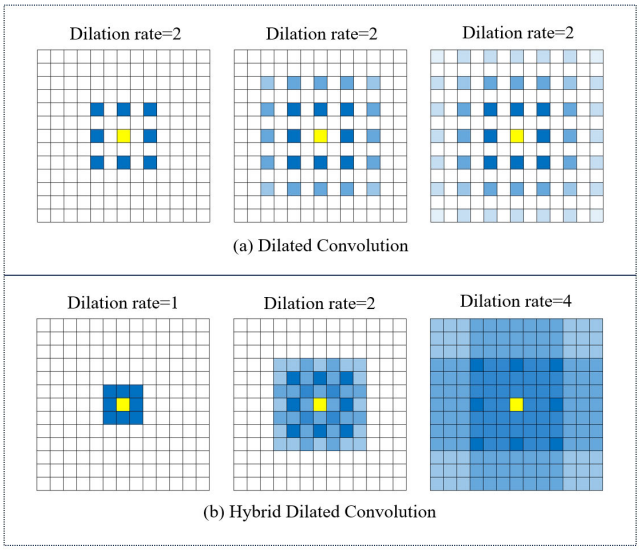

FIGURE 2. Schematic figure of dilated convolution and hybrid dilated convolution.

convolution layers while reducing computational load. The output size of the convolution can be calculated as:

$$o = \left\lfloor \frac{i + 2p - k}{s} \right\rfloor + 1$$

where i is the size of input, o is the size of output, p is the padding, k is the size of kernel, and s is strides.

However, pooling methods also reduce spatial resolution. To address this issue, dilated convolutions [31], also known as atrous convolutions, were introduced as depicted in Figure 2 (a) Dilated Convolution. This technique incorporates a "dilation rate" into the convolutional layers, enhancing the receptive field by inserting spaces within the convolution kernel. This allows the network to maintain the relative spatial positions of pixels without increasing computational complexity or the number of model parameters. The principle of dilated convolutions involves inserting zeros within the convolution kernel to expand the receptive field. When the dilation rate is set to 1, dilated convolution simplifies to standard convolution. By increasing the dilation rate, the receptive field can be effectively enlarged without adding computational burden or model parameters, thereby improving feature extraction efficiency while preserving image resolution.

Assume the kernel size is k, resulting in $k-1$ intervals within the kernel. For a dilation rate of d, each interval in the kernel needs to be padded with $d-1$ zero elements, leading to a total of $(k-1)\times(d-1)$ zero elements. This can be understood as filling the original $k-1$ intervals with $d-1$ rows (or columns) of zero elements, resulting in a new effective kernel size k':

$$k' = (k-1)(d-1) + k = (k-1)d + 1$$

Assume the kernel size is k, resulting in $k-1$ intervals within the kernel. For a dilation rate of d, each interval in the kernel needs to be padded with $d-1$ zero elements, leading to a total of $(k-1)\times(d-1)$ zero elements.

This can be understood as filling the original $k-1$ intervals with $d-1$ rows (or columns) of zero elements, resulting in a new effective kernel size k':

$$o = \left\lfloor \frac{s + 2p - [(k-1)d + 1]}{s} \right\rfloor + 1$$

In early iterations of various small to medium-sized inpainting models [13], [16], [32], the application of dilated convolution can be commonly found. However, Dilated Convolution also presents two issues. Firstly, there is the Gridding Effect. When stacking multiple identical Dilated Convolutions, numerous pixels within the receptive field remain unutilized, creating significant gaps. Consequently, this leads to a loss of continuity and completeness in the data, which hinders learning, as illustrated in Figure 2 (a) Dilated Convolution. This figure demonstrates the effect of applying a

3×3 kernel through three successive identical Dilated Convolutions. Secondly, long-ranged information might not be relevant. Dilated Convolution is designed to capture long-range information, but some of this information might be entirely unrelated to the current point, thereby affecting data consistency. Moreover, relying solely on information from large dilation rates might improve segmentation performance for large objects, but it could be detrimental for smaller objects.

Therefore, Fang et al. [33] introduced Hybrid Dilated Convolution (HDC),show in Figure 2 (b) Hybrid Dilated Convolution. Compared to conventional Dilated Convolution, HDC employs varying dilation rates across different convolutional layers. By stacking different dilation rates, HDC ensures no gaps within the receptive field. Thus, when there are n layers of dilated convolutions, the dilation rates must not share any common divisor greater than 1 to avoid the gridding effect. For the $i-th$ layer, the dilation rate r_i and the maximum dilation rate M_i must satisfy:

$$M_i = \max[M_{i+1} - 2r_i, M_{i+1} - 2(M_{i+1} - r_i), r_i$$

by default $M_n = r_n$. For instance, when the kernel size is 3, setting the dilation rates to 1, 2, and 4 over three layers of dilated convolutions allows the final layer to have a larger receptive field without losing substantial local information. This approach is illustrated in Figure 2(b), which depicts the Hybrid Dilated Convolution.

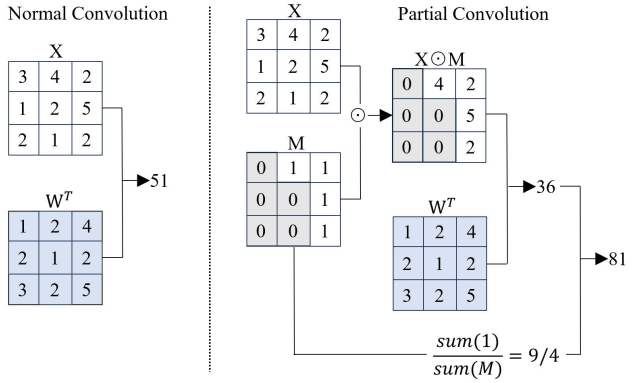

FIGURE 3. Comparison of normal convolution and partial convolution.

Based on the aforementioned theoretical analysis and extensive empirical evidence, Dilated Convolutions have been proven to be an effective technique for expanding the receptive field without increasing computational burden and have been widely adopted in deep learning. However, the application of Dilated Convolutions alone does not fully address the challenge of capturing long-range dependencies. In practical applications, it is generally necessary to combine this approach with other strategies to further enhance model performance.

b: PARTIAL CONVOLUTION

Traditional convolution treats each pixel as a valid value for computation, which is suitable for classification and detection tasks where all pixels in the input image are valid. In these tasks, standard convolution uses a sliding window approach to extract local features. However, for inpainting tasks, the images to be restored contain holes with invalid pixels that need to be distinguished from valid content. Partial Convolution [20] addresses this by incorporating a mask into the convolution operation, significantly enhancing computational efficiency and distinguishing between damaged and undamaged regions, thereby improving sensitivity to invalid pixels. The specific algorithm is shown in Figure 3 Comparison of normal convolution and partial convolution.

W represents the convolution kernel weight parameters, b indicates the corresponding bias parameters, X represents the pixel features in the sliding window corresponding to the convolution kernel, and M signifies the binary mask that determines the irregular shape. The feature data after local convolution can be represented as:

$$x' = \begin{cases} W^T(X \odot M)\dfrac{sum(1)}{sum(M)} + b, & \text{if sum(M)} > 0 \\ 0, & \text{otherwise} \end{cases}$$

where $\odot$ denotes the element-wise multiplication operation, and $sum(1)$ represents the number of pixels in the sliding window, which is the size of the convolution kernel. As shown in the formula, the feature data obtained after local convolution is related only to the unmasked input data. The term $\frac{sum(1)}{sum(M)}$ represents the scaling factor, which adjusts the variation of the effective input appropriately. When a pixel is valid, but the surrounding valid pixels are few, the convolution value at that position will be small. The scaling factor will be large in this case, amplifying the convolution data to balance the feature magnitude. The mask update rule is as follows:

$$m' = \begin{cases} 1, & \text{if } sum(M) > 0 \\ 0, & \text{otherwise} \end{cases}$$

This operation can be embedded into the forward propagation process. As long as the input image contains valid pixels, after a sufficient number of partial convolution operations, all data in the mask M will eventually be updated to 1.

Leveraging the efficacy of partial convolution, Mohite and Phadke [34] proposed a model that combines partial convolution with contextual attention to reconstruct images using new content. This model designates and compresses the available structures in the surrounding regions for the inpainting process. PPCGN [35] employs a generator with a partial convolution layer, a fully convolutional discriminator network, and a Long Short-Term Memory (LSTM) module. It achieves notable results in restoring large-hole images through a four-step process.

The introduction of Partial Convolution has significantly advanced the ability of traditional convolutional neural networks to repair irregularly damaged regions. In contrast to conventional convolution, where a substantial amount of computation is wasted when applied to damaged areas of an image due to the pixel values being zero, Partial

Convolution improves computational efficiency. Additionally, while traditional convolutional kernels cannot differentiate between damaged and undamaged areas and are insensitive to the information disparity between them, the mask update mechanism in Partial Convolution enhances inpainting performance. However, this mask update is manually defined, and although theoretically sound, it cannot guarantee accuracy in all situations, potentially introducing some errors.

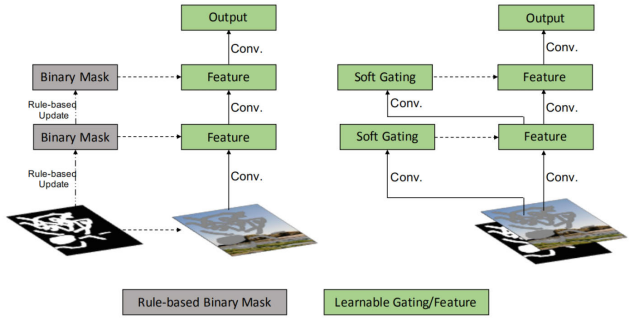

FIGURE 4. Illustration of partial convolution (left) and gated convolution [36].

c: GATE CONVOLUTION

Partial Convolution heuristically categorizes spatial positions as valid and invalid when updating the mask. Regardless of how many pixels the filter range of the previous layer covers, the mask for the next layer is set to 1, which seems somewhat unreasonable. Secondly, all channels in each layer share the same mask, which limits flexibility. Essentially, partial convolution can be viewed as a non-learnable single-channel feature hard gating. Based on these two deficiencies, gated convolution [36] discards the hard mask updated by fixed rules and instead learns a soft mask from the data, as shown in the following formula and Figure 4 Illustration of partial convolution (left) and gated convolution [36]:

$$\begin{aligned} \text{Gating}_{y,x} &= \sum\sum W_g \cdot I \\ \text{Feature}_{y,x} &= \sum\sum W_f \cdot I \\ O_{y,x} &= \phi\left(\text{Feature}_{y,x}\right) \odot \sigma\left(\text{Gating}_{y,x}\right) \end{aligned}$$

where σ represents the sigmoid activation function for output gating values between 0 and 1, ϕ is the activation functions, W_g and W_f represent two different convolutional filter.

Gated convolution enables the network to learn a dynamic feature selection mechanism for each channel and each spatial position. Visualization of the intermediate gating values shows that it can not only select features based on the background, mask, and sketch but also consider the semantic segmentation of certain channels. Even in deeper layers, gated convolution learns to highlight the mask region and sketch information across different channels to generate better inpainting results.

Gated convolution has shown excellent results in repairing irregular holes; however, compared to regular convolution, it nearly doubles the number of parameters and processing time. Therefore, Light Weight Gated Convolution (LWGC) [37] was proposed. LWGC has three variants: depthwise separable $LWGC^{ds}$, pixel-wise $LWGC^{pw}$, and single-channel $LWGC^{sc}$. $LWGC^{ds}$ employs depthwise convolution followed by 1×1 convolution to compute gating. $LWGC^{pw}$ uses pixel-wise or 1×1 convolution to calculate the gate. $LWGC^{sc}$ outputs a single-channel mask, which is broadcasted to all feature channels during multiplication. $LWGC^{sc}$ is used for all layers in the coarse network, while $LWGC^{ds}$ or $LWGC^{pw}$ is used for all layers in the fine network.

Yi et al. [37] developed a model that generates high-frequency residuals by weighted aggregation of residuals from upper and lower text samples as missing content, requiring only coarse low-resolution prediction results. Additionally, they used an attention module to compute attention scores and perform attention transfer within the U-Net structure, which improves image inpainting quality at multiple scales. Furthermore, they designed a lightweight gated convolutional network to reduce model memory usage and computation time. To enhance the descriptive power of the model, Wang et al. [38] first generated edge maps of occluded regions using prior facial knowledge. These edge maps were then used to constrain the gated convolution process, enabling precise inpainting of local features. Cao and Fu [39] employed the encoder-decoder framework to learn the sketch tensor space, enhancing the authenticity of edges, lines, and connection points during image inpainting. The model also incorporates gated convolution and an efficient attention module to improve performance. In this work, a Dense Gated Convolutional Network (DGCN) [40] is proposed by modifying the architecture of gated convolutional networks. Firstly, the Holistically-Nested Edge Detection (HED) is utilized to predict edge information of the missing regions, which assists the subsequent inpainting task and reduces artifacts and blurriness. Secondly, dense and indirect connections are incorporated into the generation network to reduce the number of network parameters and minimize the risk of instability during training. Shen et al. [41] introduced a channel attention mechanism to emphasize structural and textural features. The bidirectional gated feature fusion module ensures feature consistency, while the context feature aggregation module employs deformable convolutions to better capture image features.

Based on the above discussion, it can be concluded that Gated Convolution is a convolutional operation inspired by the Long Short-Term Memory (LSTM) network, which allows the network to selectively focus on certain features or ignore less relevant ones. This enables more flexible control of information flow, thereby reducing information loss and redundancy, and enhancing the model's ability to learn complex features. However, Gated Convolution also increases the computational complexity of the model and the number of parameters, which may lead to overfitting, particularly when

the dataset is small. In practical applications, it is necessary to balance these advantages and disadvantages according to the specific characteristics of the task and dataset to select the most appropriate model architecture.

d: REGION-WISE CONVOLUTION

In the decoder of an inpainting network, the convolutional layers apply the same filters to process features from the previous layer without distinguishing between different spatial locations. However, this approach is problematic because some locations in the subsequent layer's features correspond to valid regions of the original image, while others correspond to areas that need filling. Using the same filters for these distinct types of regions is unreasonable. Filters at valid locations should transmit and reconstruct information from the input image, whereas filters at invalid locations should generate new content based on the surrounding valid areas. Additionally, convolutional filters can only operate on local regions, failing to utilize global information beyond the filter's receptive field, which is inadequate for image inpainting.

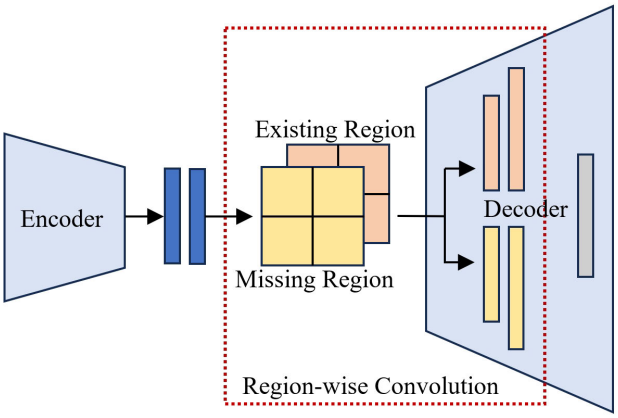

FIGURE 5. Region-wise convolution in decoder.

As shown in Figure 5 Region-wise convolution in decoder, when processing the feature map from the previous layer, two separate convolutional layers are used: one specifically handles the valid regions, while the other deals with the invalid regions. The results from these two convolutions are then integrated based on a mask. This approach is known as region-wise convolution [42]. The specific mathematical formula is as follows:

$$x' = \begin{cases} W^T x + b, & x \in X \odot M \\ \hat{W}^T x + \hat{b}, & x \in X \odot (1 - M) \end{cases}$$

In this formula, W represents the convolution filter weights for the known regions, $\hat{W}$ represents the convolution filter weights for the missing regions, b and $\hat{b}$ are the corresponding biases, X is the feature map, and x is the feature within the current convolution.

Region-wise Convolution is a widely used technique in image processing and deep learning that allows the model to perform convolution operations independently within each region. This approach is particularly useful for handling regions with distinct features or structures. Additionally, by decomposing the computational tasks across different regions of the image, it enhances the model's flexibility and efficiency. However, since Region-wise Convolution conducts operations independently in each region, this method may face challenges such as information isolation, increased computational complexity, and difficulties in parameter optimization.

e: HYPERGRAPH CONVOLUTION

CNNs (Convolutional Neural Networks) operate under the strict assumption that input data must have a regular grid-like structure. This limitation hampers their generalization and application in tasks where data often exhibits irregular structures. To address this widespread issue, there has been growing interest in Graph Neural Networks (GNNs) [43], which learn deep models using graph-structured data. Most existing methods assume pairwise relationships between objects of interest, whereas real-world scenarios are significantly more complex. Bai et al. [44] introduced hypergraph convolution within the family of Graph Neural Networks, defining the fundamental formula for performing convolutions on hypergraphs as follows:

$$x_i^{(l+1)} = \sigma\left(\sum_{j=1}^{N}\sum_{\epsilon=1}^{M} H_{i\epsilon}H_{j\epsilon}W_{\epsilon\epsilon}x_j^{(l)}P\right)$$

In matrix form, it can be expressed as:

$$\mathbf{X}^{(l+1)} = \sigma(\mathbf{H}\mathbf{W}\mathbf{H}^{\mathbf{T}}\mathbf{X}^{(l)}\mathbf{P})$$

where the dimensional relationships are $\mathbf{X}^{(l)} \in \mathbb{R}^{N\times F^{(l)}}$ and $\mathbf{X}^{(l+1)} \in \mathbb{R}^{N\times F^{(l+1)}}$. This form does not account for the numerical instability caused by convolution operations in the frequency domain and increases the risk of gradient explosion/vanishing. Therefore, proper normalization is necessary:

$$\mathbf{X}^{(l+1)} = \sigma(\mathbf{D}^{-\frac{1}{2}}\mathbf{H}\mathbf{W}\mathbf{B}^{-1}\mathbf{H}^{\mathbf{T}}\mathbf{D}^{-\frac{1}{2}}\mathbf{X}^{(l)}\mathbf{P})$$

Wadhwa et al. [45] first introduced spatial feature hypergraph convolution to learn complex relationships between data, combined with a discriminator using gated convolution to enhance structural and detail inpainting. Somani et al. [46] employed hypergraph image inpainting techniques to fill in missing information, aiming to improve the resolution of SAM images, demonstrating that combining SAM with hypergraphs can produce noise-robust interpretations. Li et al. [47] proposed a two-stage network inpainting algorithm based on U-net edge generation and hypergraph convolution to achieve reasonable structural inpainting and fine texture reconstruction in large irregular missing areas with complex backgrounds.

Based on the above discussion of Hypergraph Convolution theory, this convolution technique is capable of capturing higher-order relationships in image processing, thereby improving the efficiency of managing complex interactions among multiple nodes. It can also integrate global information, providing richer information aggregation and enabling the fusion of multimodal data for more comprehensive and efficient feature integration. Although Hypergraph Convolution has seen considerable application in image denoising, super-resolution, and deblurring [48], [49], its use in image inpainting remains limited. This may be due to its applicability and effectiveness being influenced by data characteristics, as large-scale image processing demands high computational costs. Moreover, while Hypergraph Convolution excels at emphasizing relationships, it may not yet be robust enough in constructing effective graph structures that accurately reflect image content. Consequently, not all image processing tasks are well-suited for hypergraph convolution.

2) ATTENTION LAYERS

In CNNs, convolutional layers and pooling layers effectively extract features from images, while fully connected layers map these features to categories. However, traditional CNN models do not consider the interrelationships between different features. To address this issue, the aforementioned strategies for layers introduce various mechanisms to capture long-range dependencies and contextual information in the input data, thereby enhancing the model's understanding of the data. Nevertheless, these methods have limited capabilities for extracting long-distance features, prompting many researchers to adopt global attention mechanisms.

a: CNN ATTENTION

Yu et al. [15] incorporated an attention module into their refinement network, calculating patch similarity using cosine similarity and computing a weighted sum of contextual information, transferring details from the background to the foreground. The specific operational process is illustrated in the Figure 6 Contextual attention layer. Segment the background region into patches and use these patches as convolution kernels to perform convolution operations on the foreground region, calculating the cosine distance between each foreground position and each background patch. Subsequently, compute the softmax along the channel dimension to obtain the attention values between each background patch and each foreground position. Finally, apply a deconvolution operation to achieve a weighted sum based on the attention values, thereby deriving the feature for each foreground position.

To overcome the difficulty of directly learning high-dimensional image data distribution, Song et al. [50] divided the task into inference and translation as two separate steps, connecting the two networks with a Patch-Swap module that utilizes an attention mechanism. To avoid the attention module using misleading or incorrect information, even if it comes

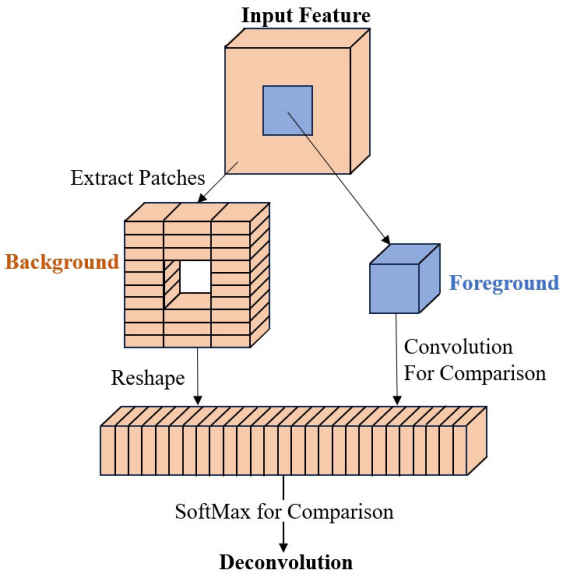

FIGURE 6. Contextual attention layer.

from the background feature map, the input and mask regions should be sufficiently large. Therefore, MUSICAL [51] places a multi-scale attention layer before the third-to-last deconvolution resolution layer, where the feature map size is 64×64. PEPSI [32], a fast and lightweight image inpainting model, also uses a contextual attention module to reconstruct fine details in images. Wang et al. [52] proposed a specialized multistage attention module to optimize the image inpainting model in a coarse-to-fine manner. Zeng et al. [53] proposed a guided upsampling network that extends the contextual attention module to borrow high-resolution feature blocks from the input image, achieving high-resolution inpainting results. The PEN-Net [21] and CRA [37] inpainting model also uses this contextual attention mechanism to extract low-level information to guide the refinement of inpainting. To understand global information and improve the realism and visual consistency of the inpainting results, UCTGAN [54] introduces a cross-semantic attention layer that leverages long-range dependencies between known and completed parts. Xie et al. [55] introduces a learnable bidirectional attention map module based on Partial Convolution for mask updating, enabling the U-Net decoder to focus on filling irregular holes. The MSAG network [56] integrates spatial attention at each scale through several multi-scale attention units to emphasize the most likely attentive spatial components, while channel attention serves as a global semantic detector, establishing connections between the multiple scales. The NLKFil [57] model proposed a single-stage network utilizing Large Kernel Attention (LKA) to handle high-resolution damaged images. LKA effectively captures both global and local details, similar to Transformer and CNN networks, thereby enabling high-quality inpainting.

In the field of image inpainting, the Contextual Attention Layer is a common technique that enhances a model's understanding of image content, helping it to learn contextual information and better restore missing areas. In image inpainting tasks, the contextual attention layer can serve as a regularization method, aiding in the prevention of overfitting and improving the model's generalization capability. For small or medium-sized images, the contextual attention layer is relatively efficient in terms of memory usage and computational cost. However, for large images, its memory and computational demands may render it impractical.

b: TRANSFORMER

The Transformer model [22], initially introduced for machine translation tasks and achieving significant advancements, has been adapted by researchers for image inpainting tasks. By leveraging the attention mechanism, it learns the global content of images, enabling the completion of large missing areas. Compared to convolutional layers, the Transformer effectively addresses the limitation of convolutional layers that can only capture local receptive fields. The detailed process is illustrated in Figure 7 Transformer of image inpainting.

First, the patch embedding step involves dividing the original input image into fixed-size patches. These patches are then fed into a linear projection (embedding layer) that flattens the patches, thereby transforming the computer vision problem into a natural language processing problem through patching and flattening. Subsequently, positional embedding is performed to add positional information to each token. This sequence is then input into the standard Transformer encoder. Finally, the output is processed through a decoder to generate the inpainted image.

Zhou et al. [23] were the first to propose using a Transformer for repairing complex scene images. Their model begins with a coarse repair by aligning the target image using the predicted depth map of the original image. They then introduced a color space transformer to achieve color and spatial matching. Finally, a fusion module combines these repair results. This method is effective for images with large missing areas and complex depth, but it is not suitable for images with low light or extreme lighting changes. Later, Wang et al. [58] developed a two-stage blind face inpainting approach. Initially, a frequency-guided Transformer detects missing areas by learning the relationships between image contexts. Then, a top-down refinement encoder-decoder architecture repairs image features hierarchically to produce semantically consistent missing content. However, this model has difficulty repairing smaller visual areas in images. MAE [24] is based on an asymmetric encoder-decoder architecture, where the encoder operates only on a subset of visible patches, while a lightweight decoder achieves excellent image inpainting results. Huang et al. [59] introduced a novel U-Net-based inpainting network, called the Sparse Self-Attention Transformer (Spa-former). Spa-former retains the advantages of transformers in long-range dependency modeling while significantly reducing computational complexity. The network employs a new channel attention mechanism that reduces the attention computation complexity to a linear scale, and replaces the traditional softmax function with ReLU to generate sparse attention maps. This design effectively eliminates irrelevant features and enhances the performance of image inpainting tasks. HINT [60], an end-to-end high-quality image inpainting Transformer, includes an innovative Mask-Prioritized Downsampling (MPD) module designed to retain visible information extracted from the damaged image while preserving the integrity of information crucial for high-level inference. Additionally, it introduces a Spatially-Activated Channel Attention Layer (SCAL), an efficient self-attention mechanism for modeling damaged images across multiple scales.

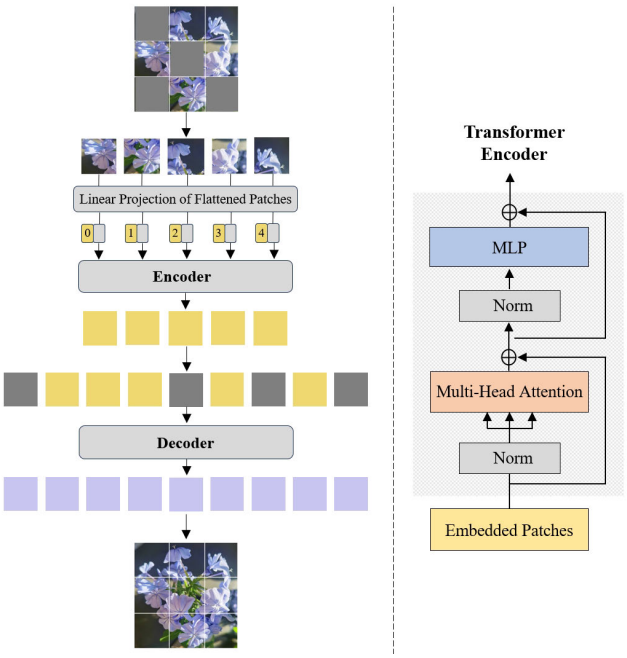

FIGURE 7. Transformer of image inpainting.

However, the global attention mechanism of Transformer requires a huge amount of computation and storage, making it difficult to achieve efficient training and fast convergence [61]. Since CNNs have excellent performance in texture feature extraction, a series of studies have combined the two, proposing more hybrid models. Zheng et al. [62] used a convolutional network with small and non-overlapping receptive fields (RF) for weighting, allowing the Transformer to model long-range visible contextual relationships of equal importance at all layers without implicitly mixing them when using larger RFs. To enhance the appearance consistency between visible and generated regions, they introduced a new Attention-Aware Layer (AAL) to better utilize distant, highly correlated high-frequency features. Dong et al. [25] proposed a hybrid model that first uses a powerful attention-based transformer model in a fixed low-resolution sketch space to

restore the overall image structure, providing more reliable structural information for the subsequent CNN inpainting network. Transinpaint [63], a context-adaptive transformer for image inpainting, first restores a low-resolution image using a Transformer network. This low-resolution image is then upsampled and further refined through a convolutional and Transformer hybrid network to achieve the final inpainting. ZITS [25] designs an incremental Transformer structure inpainting network, which uses mask position coding to improve the generalization of the model to different masks, a Transformer structure restorer to restore the image structure, a structural feature coder to encode the structural features of the image, and a Fourier CNN texture restorer to restore the texture information of the image, respectively. These designs can improve the performance of the model in restoring large missing regions. ZITS++ [64] is an improved model of ZITS The overall structural prior is recovered on low-resolution images by using the Transformer Structure Restorer (TSR) module, and then further upsampled to higher resolution images by the Simple Structure Upsampler (SSU) module. Finally the upsampled structural prior from TSR is further processed using Structure Feature Encoder (SFE) before the original Fourier CNN Texture Restoration (FTR) module and incrementally optimized using Zero-initialized Residual Addition (ZeroRA) for incremental optimization to improve the model performance. CMT [65] uses a continuous mask to represent the amount of error in a token. First, it is used in self-attentive repair networks to update the masks according to the initialization mask, which is propagated through several masked self-attention and mask update (MSAU) layers to achieve good repair. Zhou et al. [66] developed a novel self-supervised attention-based generative adversarial image inpainting method, leveraging the self-attention mechanism of Transformers to capture global semantic information. This approach overcomes the limitations of traditional convolutional operations by introducing self-supervised attention modules within the Transformer. Concurrently, the discriminator employs a hierarchical Swin Transformer with a shifted window strategy to extract contextual features of the image, while the generator utilizes depthwise over-parameterized convolution layers (DO-Conv) to enhance model performance. The Structure-guided Synergistic Transformer (SyFormer) [67] employs a dual-routing filtering module that uses a progressive filtering strategy to eliminate irrelevant noise interference and establish global texture correlations. Simultaneously, a compact perception module maps affinity matrices within the structure priors introduced by the structure-aware generator, facilitating the matching and filling of corresponding patches in extensively damaged images. The Inpainting Transformer (ITrans) [68] network integrates global and local transformers with convolution operations to enhance the convolutional encoder-decoder structure. This combination allows for global relationship modeling and local detail encoding, which are crucial for generating realistic hallucinated images.

In addition to the aforementioned models that generate images from scratch, a series of models based on the Denoising Diffusion concept, which extract images from noise through denoising, such as DDPM [26], DDIM [27], and IDDPM [69], have employed hybrid layers combining CNN and Transformer architectures to achieve enhanced performance.

The transformer has a natural advantage in handling long-range dependencies due to their self-attention mechanism, which can capture contextual information across any distance. They are well-suited for processing large images, as they can be parallelized and typically do not pose significant memory challenges. However, Transformers may require substantial computational resources, and their large number of parameters necessitates extensive datasets to prevent overfitting.

3) SUMMARY OF LAYERS

Deep learning-based image inpainting networks primarily rely on convolution and attention layers. Each model has its own advantages, leading to different application scenarios. With the support of large-scale image datasets, massive models with a vast number of parameters, whether CNN-based or Transformer-based, can achieve excellent results in image inpainting after training. However, in fields where obtaining large amounts of training images is challenging, such as medical image inpainting and cultural heritage inpainting, CNNs benefit from their translational invariance and fast convergence during training, leading to broader applications. Nevertheless, small models solely relying on CNNs often struggle to achieve highly satisfactory results. Consequently, many researchers now opt to use large pre-trained models as a foundation, further fine-tuning them with domain-specific image data to achieve better outcomes.From a layered perspective, on one hand, an increasing number of models are integrating Transformer and CNN architectures to achieve more precise and rational inpainting effects. On the other hand, algorithms from other domains, such as hypergraph convolution [44], are gradually being incorporated into image inpainting algorithms, introducing new ideas and approaches to the field.

B. CONNECTIONS

In deep learning networks, connections determine the pathways for data flow, thereby dictating the transmission of information and features, which is crucial for the model's performance. Given that the overall structure of image inpainting models primarily consists of two modules, the encoder and the decoder, we refer to the connections between these two modules as inter-module connections, and the connections within a single module as intra-module connections. The following sections will provide a detailed introduction to the strategies applied to both inter-module and intra-module connections.

1) INTER-MODULE CONNECTIONS

The generator in deep learning-based image inpainting models can primarily be divided into an encoding part and a decoding part. If the decoder's source of information is solely the output of the final layer of the encoder, we refer to this as a standard encoder-decoder structure. However, if the decoder's source of information includes not only the output from the final layer of the encoder but also the intermediate feature maps from each corresponding layer of the encoder, we refer to this as a U-Net structure, as illustrated in Figure 8(a) Inter-module Connection.

a: ENCODER-DECODER

The Encoder Decoder is a model structure derived from the Auto Encoder [70], as illustrated in Figure 8(a)① Encoder Decoder. The encoder compresses the input data into a latent space representation, while the decoder reconstructs the acquired features and outputs them. This model can effectively utilize the known information of the image to generate content that closely resembles the original image, making it highly effective for inpainting tasks. The simplicity of the Encoder-Decoder model structure has led to extensive research and application in this area.

The most classic method in the encoder-decoder category is the Context Encoder (CE) model proposed in 2016 [12]. This model is an unsupervised semantic feature inpainting approach that combines the concept of GANs to generate content for arbitrary regions of an image based on the context surrounding the missing areas. However, in CE, the adversarial loss is applied only to the missing regions, resulting in discontinuities and structural inconsistencies between the missing and intact regions. To address this issue, Iizuka et al. [13] proposed the GLCIC model, introducing both global and local discriminators for image inpainting. Yeh et al. [71] aimed to improve semantic image inpainting by searching for the closest encoded representation of the image to be restored from the latent space, and then using this encoding to reconstruct the image with a generator. The StackGAN model [72] proposed a two-stage coarse-to-fine inpainting process. The GMCNN model [14] used convolutional kernels of different scales to convolve the image, obtaining features under different receptive fields. By applying global and local loss constraints, they used an implicit diversified Markov Random Field to diffuse the predicted structural information into the missing regions. Liao et al. [73] proposed an edge-based contextual decoder, which first extracts edge information from the image, then uses a fully convolutional network to restore the edges in the missing regions, and finally applies a contextual encoder to repair the restored edge information and the incomplete image. Song et al. [74] introduced semantic segmentation information into image inpainting. By separating inter-class and intra-class differences, they achieved clearer boundary inpainting between semantically different regions and more accurate texture within semantically consistent regions. To enhance the structural coherence of restored images, Vo et al. [75] proposed a model that performs structural inpainting of various visual scenes through a two-stage training process. The first stage introduces structural loss to constrain the model, while the second stage uses adversarial loss to optimize the model structure. Yu et al. [36] replaced traditional convolutions with gated convolutions, which dynamically select features for each channel and spatial location in the image, learning feature maps that adapt

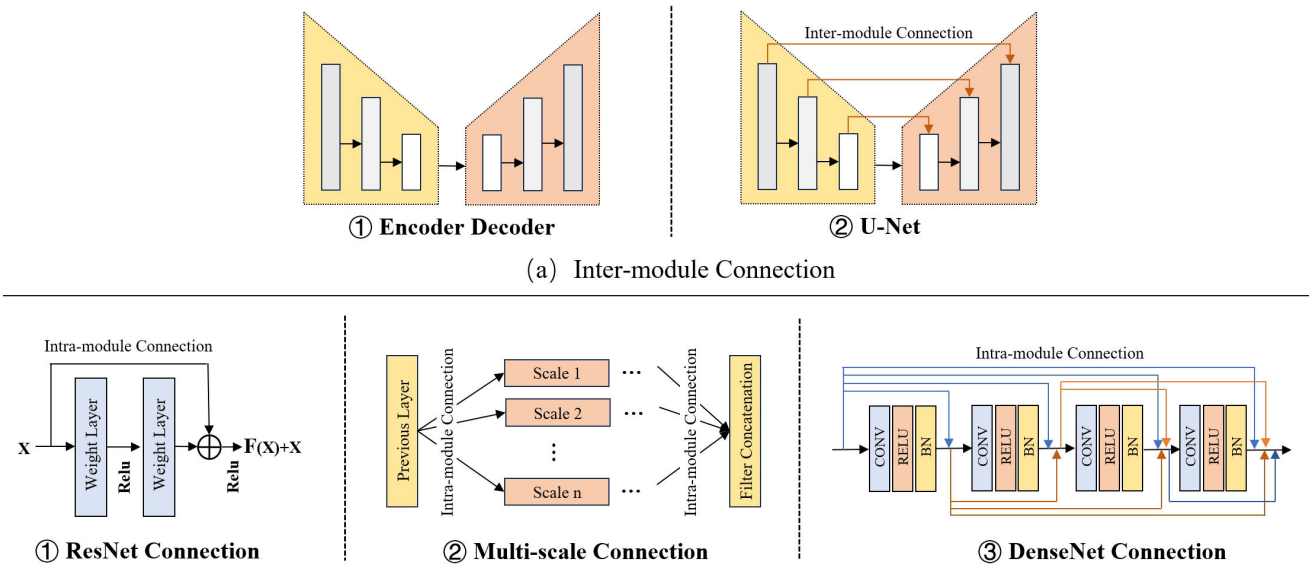

FIGURE 8. Transformer of image inpainting.

to background, mask, and sketch outlines drawn through human-computer interaction, leading to better inpainting results. The PEPSI network [32], which unifies a two-stage cascaded network into a single-stage encoder-decoder network. This network consists of a single encoder and two parallel decoders. The two parallel decoders provide coarse and fine inpainting paths. The coarse inpainting path generates coarse results based on the feature maps from the encoder, while the fine inpainting path first reconstructs the feature maps via CAM and then uses the decoder to generate higher-resolution feature maps. He et al. [24] proposed a Transformer-based image inpainting model, Masked Autoencoders (MAE), which uses an encoder-decoder structure. This model allows self-supervised learning in computer vision.

b: U-NET

The U-Net model [17] is highly effective for image inpainting due to its unique feature fusion capability. As shown in Figure 8(a)② U-Net, U-Net combines features of different scales along the channel dimension. The input to each layer of the decoder module includes not only the output from the previous layer but also the corresponding scale layer's output from the encoder module through inter-module connections. U-Net ensures that the results are derived from contextual features, allowing the encoder to constantly constrain the decoder, thus significantly reducing loss and improving model effectiveness.

The Shift-Net model [18] enhances the U-Net architecture by introducing a shift connection layer to fill arbitrary-shaped defective areas with complex structures and fine textures. Liu et al. [20] proposed partial convolution to replace all convolution layers in U-Net and used nearest-neighbor upsampling during the decoding phase to repair irregular defects. The DFNet model [19] integrates multiple fusion modules into the last five decoding layers of U-Net to eliminate the discontinuities that arise when blending defective and intact parts. The Pyramid Context Encoder Network (PEN) model [21] encodes contextual semantics from full-resolution input and decodes the learned semantic features for inpainting. During training, the encoder gradually learns the correlations between regions from high-level semantic feature maps using attention and transfers these correlations to low-level feature maps, ensuring visual and semantic consistency in image inpainting. Yi et al. [37] developed a model that generates high-frequency residuals by aggregating the residuals of contextual samples weighted to fill in the missing content, requiring only rough low-resolution prediction results. They also employed an attention module to calculate attention scores and perform attention transfer within the U-Net structure, enhancing image inpainting quality across multiple scales. Additionally, they designed a lightweight gated convolutional network to reduce memory usage and computation time. Wang et al. [56] enhanced U-Net by adding multi-scale attention modules, which compute both low-level detail similarity and high-level semantic similarity of images. The second stage of Transinpaint [63], the TENet inpainting network, builds upon U-Net by adding a Feature Synthesis Module to the decoding layers. This module employs a hybrid network of convolution and Transformer mechanisms, enabling a more comprehensive and robust representation of image content. ISFRNet [76] is a three-stage network designed for high-quality image inpainting. The first stage utilizes a pre-trained pSp-StyleGAN model, which generates highly realistic facial images with rich structural features. The second stage employs a shallow network with a small receptive field to capture fine details. The final stage features an enhanced U-Net architecture with two encoders and one decoder, providing a large receptive field for more comprehensive feature extraction and synthesis. Dong et al. [77] proposed an improved dual-stream U-Net algorithm by incorporating attention mechanisms into two U-Net networks, referred to as the Dual AU-Net network, to enhance image texture details. Additionally, location codes (LC) of the damaged regions are included to guide the network in repairing and accelerating convergence. The generator in the adversarial network employs a Least Squares GAN (LSGAN) loss to capture more content details and improve training stability. In a series of models based on the denoising diffusion concept, such as DDPM [26], DDIM [27], and IDDPM [69], U-Net has been used as the fundamental architecture, achieving excellent results in both image inpainting and generation.

2) INTRA-MODULE CONNECTIONS

Intra-module connections can enhance model performance by increasing both the depth and width of the model. To address the degradation problem in deep networks when increasing depth, residual connections [7] are employed. To increase model width, many models utilize multi-scale approaches. Additionally, DenseNet [78] improves the network by employing numerous connections based on the assumptions of information flow and feature reuse.

a: RESIDUAL CONNECTIONS

To address the degradation problem in deep networks, one can manually skip connections over certain layers of neurons, connecting non-adjacent layers and thereby weakening the strong dependency between successive layers. This type of neural network is known as a Residual Network (ResNet) [7]. The ResNet paper introduced the residual connection to mitigate the degradation problem, as illustrated in Figure 8(b)① Residual Connection. By adding the input to the output through intra-module connections, Residual Connections help address the vanishing and exploding gradient problems.

Full Resolution Residual Networks (FRRN) [79] assist in feature integration and texture prediction to fill irregular holes through a residual architecture. Naive upsampling of low-resolution inpainting results often leads to blurred images; however, incorporating high-frequency residuals can produce detailed and sharp images. Based on this

observation, Yi et al. [37] proposed a Contextual Residual Aggregation (CRA) mechanism. This mechanism generates high-frequency residuals by weighted aggregation of residuals from contextual patches, relying solely on low-resolution predictions. Yang et al. [80] introduced a novel coarse-to-fine residual inpainting framework that first reconstructs a downsampled low-frequency coarse outline at a lower computational cost, followed by the filling of high-frequency details. These details are then added as residuals to the coarse outline, thereby better preserving the structural and textural details in the synthesized results. SRPNet [81] proposed a Semantic Residual Pyramid Network based on deep generative models.

b: MULTI-SCALE CONNECTION

Increasing network width through parallel multi-scale connections can enhance the quality and efficiency of image inpainting. Multi-scale networks, as shown in Figure 8(b), can capture different receptive fields using convolution kernels of varying sizes, thereby extracting features of different scales. The most classic multi-scale connection model is the series of Inception networks [28], [29], [30].

Inception V1 [28] applies convolutions with kernels of sizes 1, 3, and 5, along with max pooling, simultaneously to the previous layer, concatenating the results of these four operations as the output. Since max pooling does not alter the number of input channels, the number of channels significantly increases after several Inception modules. To reduce the computational load, a 1×1 convolution is added in each branch to decrease the number of feature map channels. Inception V2 and V3 [29] further improved computational efficiency and minimized feature information loss by decomposing large convolution kernels into smaller ones: a 5×5 convolution is replaced by two 3×3 convolutions, and an $n \times n$ convolution is decomposed into two asymmetric convolutions, $1 \times n$ and $n \times 1$, saving parameters. Inception V4 and Inception-ResNet [30] combined Residual blocks with Inception modules, achieving further performance improvements.

Additionally, Yang et al. [82] proposed a multi-scale neural patch synthesis method based on joint optimization of image content and texture constraints. This method preserves contextual structure while generating high-frequency details by matching and adjusting patches with the most similar mid-level feature correlations in a deep classification network. GMCNN [14] achieved final image inpainting by generating results at three different scales through parallel multi-scale convolutional layers and then fusing them. The MUSICAL model [51] employed a multi-scale image context attention learning strategy, which flexibly handles richer background information while avoiding misuse. To explore the local spatial components under different receptive fields and the interrelationships between multi-scale feature maps, the Multi-Scale Attention Network (MSA Net) [56] introduced a multi-scale attention group (MSAG) comprising multiple multi-scale attention units (MSAU) for a comprehensive analysis of features ranging from shallow details to high-level semantics. In each MSAU, an attention-based spatial pyramid structure was designed to capture deep features with various receptive fields.

c: DENSE CONNECTION

DenseNet [78] divides the network into several densely connected blocks. Within each dense block, every layer is connected to all preceding layers, not just the immediate one. This connectivity allows for more rapid information flow and enables the network to combine features from different layers early on. Each layer stacks its feature maps with those from previous layers, forming dense feature maps. These dense connections help mitigate the vanishing gradient problem, as each layer can directly access gradient information from earlier layers, leading to more stable training. Furthermore, since every layer connects to all preceding layers, the network can automatically learn richer and more complex feature representations. This reusability enhances the network's performance and reduces the number of parameters that need to be trained.

DenseNet is often employed in the field of image super-resolution. To address potential pixel artifacts or visual inconsistencies in image inpainting, Ma et al. [40] proposed a novel Dense Gated Convolutional Network (DGCN) by modifying the gated convolutional network structure. By guiding edge inpainting and incorporating dense connections in the generative network, the approach reduces network parameters and minimizes the risk of instability during training.

3) SUMMARY OF CONNECTIONS

The essence of connections lies in modifying the network structure to control the flow of information, thereby ensuring its quality. In a single network, inter-module connections allow multiple modules to share information, leading to the generation of better-restored images under the guidance of more comprehensive information. The encoder-decoder architecture, one of the earliest deep learning-based structures for image inpainting, is straightforward, with a clear and efficient data flow. The later U-Net architecture introduced additional data flows, helping to retain the encoder's content within the decoder, resulting in greater stability, albeit at the cost of increased complexity. Theoretically, for simple image inpainting tasks, the encoder-decoder structure can efficiently accomplish the task. However, for more complex and larger tasks, the U-Net structure offers enhanced stability. This conclusion, however, is not absolute. Within these two foundational architectures, different intra-module connections can be employed to further improve model performance. These include Residual connections, which enhance the network's depth, multi-scale connections, which increase the model's width, and Dense connections, which increase the density of data flow. Each of these connection types can enhance the model's learning capacity to some extent.

C. MULTI-NETWORKS

A single image inpainting network, when enhanced by a series of strategies, can achieve satisfactory results. However, it often falls short when dealing with complex images. To address this limitation, many researchers have proposed multi-network models. Next, we will introduce composite multi-network models from both the data usage and structural perspectives.

1) DATA PERSPECTIVE

In addition to the structure of the model itself, data quality is crucial for performance. High-quality data can help the model quickly acquire feature extraction capabilities. When dealing with complex problems, the model's learning capacity can be limited. To address this, some research efforts have focused on reducing the learning difficulty by breaking down complete images into more manageable data segments for step-by-step learning. Generally, methods for data decomposition are based on separating different frequency bands. High-frequency features correspond to short-distance features, while low-frequency features correspond to long-distance features. Thus, frequency-based step-by-step learning allows the model to learn features at different distances incrementally, significantly reducing the problem's complexity. The following sections will introduce models that employ frequency-based learning in detail.

a: FREQUENCY-BASED NETWORKS

In 2018, the Edge-Aware CE model [73] utilized the Holistically-Nested Edge Detection (HED) [83] model to extract high-frequency edge information for training. The model first repaired high-frequency edge information and then proceeded to restore the entire image. The Progressive Visual Structure Reconstruction (PRVS) network [84] introduced a novel Visual Structure Reconstruction (VSR) layer. This layer interweaves the reconstruction of visual structures and visual features, optimizing the image inpainting process through shared parameters. The EdgeConnect [16] model utilizes the Canny [85] algorithm to extract edge information from images, leveraging this high-frequency information to guide a three-step training process. First, it repairs the edges, then it uses the restored edges to reconstruct the entire image, and finally, it performs joint training of both networks. Xiong et al. [86] proposed a foreground-aware image inpainting system that clearly distinguishes between structural inference and content inpainting. The model first predicts the foreground contours, which are high-frequency information, and then performs region repair based on these predicted foreground contours. Shao et al. [87] introduced a two-stage image inpainting network based on using edges as high-frequency structural information and color as a combination of high and low-frequency information. Yang et al. [88] used Sobel [89] filters to obtain high-frequency information, incorporating this high-frequency information into the different scale feature layers restored by the decoder to impose constraints. Yi et al. [37] proposed the Contextual Residual Aggregation (CRA) mechanism, which first generates a low-resolution inpainted image using a generator. Then, high-frequency residuals are obtained through the residual aggregation module. Finally, the high-frequency residuals are combined with the low-resolution inpainted result to obtain a high-resolution inpainted image. Cao et al. [39] first used the Canny algorithm to obtain the high-frequency information of the image, training a network with this data. Then, they employed a pyramid network to generate the Sketch Tensor Space, ultimately producing the final restored image. Roy et al. [90] proposed a frequency-based deconvolution module that enables the network to selectively reconstruct high-frequency components while learning the global context. Traditional frequency-guided models typically learn frequency components and the full image in separate steps. However, Guo et al. [91] noted that texture and structure information influence each other during the inpainting process and should therefore be mutually reinforcing. Consequently, in the second stage of their network, they introduced Bi-directional Gated Feature Fusion (Bi-GFF) and Contextual Feature Aggregation (CFA) to assist image inpainting by leveraging edge information. To enhance the descriptive power of the model, Wang et al. [38] first generated edge maps of occluded regions using prior facial knowledge. These edge maps were then used to constrain the gated convolution process, enabling precise inpainting of local features. Wang et al. [58] first used a frequency-guided Transformer network to detect missing regions in images by learning the contextual relationships within the image. Subsequently, they performed a top-down hierarchical inpainting of the image features. Yamashita et al. [92] first utilized a Transformer-based model to infer high-frequency structural information, such as edges and line drawings. They then incrementally injected this structure into a CNN-based texture inpainting network using zero-initialized residual connections. Ren et al. [93] employed a low-frequency image that preserves edge information [94], [95] as guidance for structural reconstruction. This approach allows the first stage of the network to focus on restoring the global structure without interference from irrelevant texture information. The second stage then focuses on the complete inpainting of the image. The Joint Prediction Filtering and Generation Network (JPGNet) [96] utilizes PFUNet to apply the input image to a pixel sequence kernel based on filtering for inpainting. This process guides the subsequent series of inpainting tasks. Wu et al. [97] proposed a method where a network first predicts the Local Binary Pattern (LBP) [98] information of the missing regions, which then guides the actual image inpainting task in the second network. The primary reason for choosing LBP in the first network is that, compared to other options (e.g., edges), LBP contains richer structural information. SRInpainter [99] begins with global structure inference from low-resolution inputs and progressively refines local textures in the high-frequency space, forming a multi-stage framework with

super-resolution (SR) supervision. The initial stage provides structural information as a coarse SR result, serving as an appearance prior. This result is then combined with the higher-resolution damaged image in the subsequent stage to render usable textures for the missing regions. In this work, a Dense Gated Convolutional Network (DGCN) [40] is proposed by modifying the architecture of gated convolutional networks. Firstly, the Holistically-Nested Edge Detection (HED) is utilized to predict edge information of the missing regions, which assists the subsequent inpainting task and reduces artifacts and blurriness. Secondly, dense and indirect connections are incorporated into the generation network to reduce the number of network parameters and minimize the risk of instability during training. DF3Net [100] is an image inpainting model that integrates high-frequency and low-frequency features. It is based on a Hierarchical Atrous Transformer (HAT) and a Dual-Frequency Convolution (DFC) module, enabling effective fusion of these features.

b: COARSE-TO-FINE NETWORKS

Yang et al. [82] proposed a neural network-based model that first performs a coarse inpainting, followed by a trained classification network that refines the patches for detailed inpainting. This approach generates images with fine details, ensuring higher quality inpainting. Yu et al. [15] proposed a two-stage network comprising a simple coarse inpainting network trained with reconstruction loss and a refined inpainting network based on contextual attention. The coarse network provides an initial inpainting, which is then refined by the attention-based network to enhance the final image quality. Following this, they [36] proposed a two-stage image inpainting model that initially uses a coarse inpainting network based on gated convolution. This is followed by a fine inpainting network embedded with contextual information to complete the image inpainting process. Zeng et al. [101] proposed a Cr-fill model similar to the aforementioned structure. They argued that the Contextual Attention (CA) layer could only find the most similar patches but lacked direct supervision signals to ensure semantic consistency. Therefore, they removed the CA layer and introduced a reconstruction loss (CR) to address this issue. Song et al. [50] proposed a deep learning image inpainting network that first performs a coarse inpainting, followed by a patch-swap phase, and finally a fine inpainting phase. This sequential approach allows the network to gradually improve the quality and details of the repaired image. M Sagong et al. [32] proposed a model consisting of a shared encoding network and a parallel decoding network with both coarse and refinement paths. The coarse path generates an initial rendering result, which is then used to train the encoding network to predict features for the contextual attention module (CAM). Simultaneously, the inpainting path utilizes the refined features reconstructed by the CAM to create higher quality restored images. Zhou et al. [23] were the first to propose using a Transformer for inpainting images of complex scenes. The model begins by aligning the target image based on the predicted depth map of the original image to achieve coarse inpainting. Subsequently, the authors designed a color space transformer to match the color and spatial characteristics of the image. Finally, a fusion module is employed to merge the inpainting results obtained from the previous steps. To achieve high-fidelity image detail inpainting, Zheng et al. [62] proposed a two-stage approach. In the coarse inpainting stage, they introduced Restrictive Convolutional Blocks (RCBs) to extract tokens. In the fine inpainting stage, they proposed a novel Attention-Aware Layer (AAL) that adaptively balances the attention between visible content and generated content. Xu et al. [102] proposed a two-stage model for image inpainting. The process involves a patch-based assistance method, beginning with a coarse inpainting stage followed by a fine inpainting stage. This approach ensures initial broad corrections that are subsequently refined for higher accuracy and detail. Quan et al. [103] utilized a network with skip connections to perform an initial coarse inpainting. Following this, they employed a shallow-deep model with a small receptive field for local refinement. Finally, they introduced an attention-based encoder-decoder network with a large receptive field to achieve global refinement. Feature Fusion and Two-Step Inpainting (FFTI) [104] consists of three stages. First, a Dynamic Memory Network (DMN+) is used to fuse external and internal features of the incomplete image to generate an optimized intermediate image. Second, a Generative Adversarial Network (GAN) with gradient penalty constraints performs coarse inpainting on the optimized image, producing a coarse inpainted result. Finally, correlated feature consistency is applied to further refine the coarse inpainted image, resulting in a finely inpainted final output. Qu et al. [105] proposed a coarse-to-fine networks called structure-first and detail-later image inpainting workflow that uses a pyramid generator made up of several sub-generators to achieve high-quality image completion. In this method, the lower levels of the pyramid focus on reconstructing the image's overall structure, while the higher levels enhance fine details to complete the inpainting process. Jain et al. [41] proposed a novel inpainting network that combines the advantages of a coarse-to-fine GAN-based generator network, which typically generates better structural features, with state-of-the-art high-frequency fast Fourier convolution layers. This design excels in both structural generation and repetitive texture synthesis, achieving outstanding visual quality.

c: SEMANTIC-BASED NETWORKS

Song et al. proposed a SPGNet [74] model which is a two-step image inpainting process consisting of segmentation prediction (SP-Net) and segmentation guidance (SG-Net). First, the model predicts segmentation labels for the areas to be repaired, and then it generates the inpainting results

based on these segmentation guidelines. Liao et al. [106] proposed a model with a similar basic premise to SPGNet, focusing on obtaining a complete semantic segmentation map of the missing image and then using this map to guide the image inpainting process. Compared to the original SPG model, Liao et al.'s approach emphasizes mutual enhancement between semantic segmentation and image inpainting. It introduces a confidence measure for each level of semantic segmentation and avoids ambiguous segmentation by maximizing segmentation confidence. This method ultimately yields a more complete inpainting result. Liu et al. [107] identified that the blurring of textures and distortion of structures in inpainting results were due to the neglect of semantic relevance and feature continuity in missing regions. To address this, they proposed a method based on a deep generative model with a coherent semantic attention (CSA) layer. This approach not only preserves the contextual structure but also models the semantic relevance among the features in the missing regions, enabling more effective prediction of the missing parts. Ardino et al. [108] proposed a method that utilizes semantic segmentation to model the content and structure of images, learning the optimal shapes and positions for object insertion. To produce reliable results, they designed a novel decoder that combines semantic segmentation with generative blocks. This combination is intended to better guide the generation of new objects and scenes, ensuring semantic consistency with the image. Qiu et al. [109] designed the Semantic-SCA model, which includes a semantic structure reconstructor and a texture generator. First, they train the semantic structure reconstructor using a semantic structure map based on unsupervised segmentation. Then, they use the Spatial-Channel Attention (SCA) module to obtain fine-grained textures. The SCA module enhances the functionality by capturing information from long-range pixels and different channels within the model. Xie et al. [110] divided the complex image inpainting process into two main stages: semantic reconstruction and appearance synthesis. This separation approach effectively simplifies the understanding of features, thereby streamlining the model training process.

d: OTHERS

In their approach, Wang et al. [111] employed a model that first addresses the inpainting of the monochrome image, before tackling the inpainting of the full color image with a multi-scale networks. The model proposed by Zeng et al. [53] employs an iterative method where high-confidence pixels within holes are prioritized and filled in each iteration, with subsequent iterations addressing the remaining pixels, leading to gradual improvement in the inpainting results. Lahiri et al. [112] first pre-trained a GAN to map the noise distribution to various real images. Then, they trained a noise prior prediction network to predict the optimal noise from a given masked image. Finally, through iterative inference, they obtained the best matching noise prior for the given masked image and used the pre-trained GAN generator to reconstruct the image. Quan et al. [103] proposed a three-stage inpainting framework focusing on the receptive field. First, a global inpainting is performed using a large receptive field. Next, a local inpainting is conducted using a small receptive field. Finally, attention is integrated to achieve the overall inpainting. Zhang et al. [113] designed an innovative framework that combines deep learning and traditional methods. First, an existing deep inpainting model, LaMa, is used to reasonably fill in the gaps, establishing three guiding images composed of structure, segmentation, and depth. Multi-guided PatchMatch is then applied to generate eight candidate upsampled inpainting images. Kim et al. [114] applied super-resolution to the roughly reconstructed output to refine it at a high resolution, and then scaled the output back to the original resolution. By introducing high-resolution images into the refinement network, finer details can be reconstructed. Transinpaint [63], a context-adaptive transformer for image inpainting, first restores a low-resolution image using a Transformer network. This low-resolution image is then upsampled and further refined through a hybrid network to achieve the final inpainting.

2) STRUCTURE PERSPECTIVE

From a model structure perspective, as the complexity of problems increases, a single linear network structure becomes insufficient to achieve research goals. Consequently, many researchers opt for composite network structures involving multiple networks to enhance model performance. Multi-network models can be categorized into cascaded and parallel network structures. Cascaded structures require the completion of one stage before the next can commence, typically consisting of multiple complete networks. These cascaded networks generally undergo step-by-step training. In contrast, parallel structures allow two or more networks to operate simultaneously, with their results converging and integrating before proceeding to the next step.

a: PARALLEL NETWORKS

PEPSI [32] first compresses features using an encoder that includes dilated convolutions, and then decompresses and extracts these features through two parallel decoders. One decoder performs a coarse inpainting of the image, while the other decoder, utilizing an attention module, refines the image. Finally, the two results are combined to obtain the final restored image. Li et al. [84] proposes a parallel network approach where one network is responsible for structural reconstruction and the other for image reconstruction. By progressively feeding structural features into image features, the reconstruction of visual structure and visual features are intertwined, benefiting from each other through shared parameters. VCNet [115] also proposes a parallel network approach for image inpainting. The key difference is that one network is used to predict the mask, with its encoder output being fed into the decoder of another image inpainting

network. Yang et al. [88] proposed an image inpainting network that compresses image features and high-frequency features through an encoding network named Spatial Context Encoder. This is followed by parallel decoding networks for image inpainting. The primary decoder at different scales is constrained by a parallel high-frequency feature decoding network. The Joint Prediction Filtering and Generative Network (JPGNet) [96] consists of three branches. First, the prediction filtering network and the uncertainty network operate in series, followed by a deep generative network that runs in parallel with these two networks. Finally, the outputs of the three networks are fused to obtain the final image inpainting result. Guo et al. [91] proposed a method for image inpainting by running an edge repair network and an image repair network in parallel. In this approach, the encoder of the edge generator inputs features at different scales to the corresponding layers of the image inpainting network's encoder. Simultaneously, the decoder of the image inpainting network transmits features at different scales to the corresponding layers of the edge repair network's decoder for further constraints. The final restored image is obtained by merging the results from both networks. Xie et al. [99] proposed a network for repairing comics, which divides the complex inpainting process into two main stages: semantic repair and appearance synthesis. In the semantic repair network, two decoders operate in parallel. In the second stage, the appearance synthesis involves two encoders running in parallel.

b: CASCADED NETWORKS

In cascaded multi-network models, the tasks at different stages usually have distinct objectives. However, these networks leverage the inter-task correlations to enhance the model's performance. This strategy is commonly adopted in most multi-network models. For instance, models such as [15], [16], [23], [25], [36], [37], [45], [50], [53], [58], [62], [73], [74], [82], [86], [87], [90], [92], [93], [97], [101], [102], [104], [105], [107], [108], [109], [111], [112], [114], and [116] employ two cascaded networks for training. On the other hand, [39], [99], [103], [113], [117] utilizes a three-stage network for training. While three-stage training can potentially yield better results, it incurs higher training costs and longer time requirements, so the choice and design should be based on specific practical needs.

Three-stage training models generally achieve better outcomes due to their thorough approach in addressing different aspects of the image inpainting task. However, due to the increased training costs and longer time requirements, it is crucial to balance the benefits against the practical constraints and design requirements when choosing the number of stages for cascaded multi-network models.

3) SUMMARY OF MULTI-NETWORKS

When addressing complex image inpainting tasks, a single network may sometimes struggle to deliver satisfactory results. Consequently, an increasing number of models are adopting multi-network strategies to decompose a complex problem into multiple, relatively simpler subproblems. This approach leverages the capabilities of composite networks to achieve superior outcomes.

From a data perspective, the essence of multi-network strategies lies in transforming complex image data into simpler forms. A common technique involves frequency-based inpainting, where the full-spectrum image, rich in information, is decomposed into partial frequency bands with more uniform data characteristics, thereby simplifying the problem. A typical example of this is edge-guided methods, where high-frequency information, such as edges, is isolated to guide the inpainting process. The coarse-to-fine inpainting strategy, on the other hand, involves using an initial network to recover coarse information, followed by one or more subsequent networks to refine the details. In some coarse-to-fine models, the initial stage focuses on restoring low-frequency information, while the fine inpainting phase addresses high-frequency details. Other models may first employ a rapidly converging network to produce an approximate result, which is then refined by a more precise network to enhance details and structure. Semantic-based inpainting generally begins with a pre-trained semantic segmentation network, which simplifies the semantically complex image into a format that is easier for the network to interpret, allowing the model to learn effectively before generating the final image. In certain models, the concept of structural information typically refers to large regions of low-frequency data, with a smaller portion pertaining to high-frequency edge structures, and occasionally to semantic structures derived from segmentation maps.

From the perspective of network architecture, multi-network systems can be categorized into parallel and cascaded configurations. Parallel networks increase the model's width and often facilitate data exchange through inter-network connections when operating concurrently. This exchange, coupled with mutual guidance and constraint among the networks, can enhance the overall performance of the model. However, parallel networks typically result in increased network size, thereby raising computational costs. Cascaded networks, in contrast, connect multiple networks sequentially, effectively increasing the model's depth and allowing the completion of multiple sub-tasks in a stepwise manner. Serial networks significantly improve model performance and are the predominant approach in multi-network systems, though they generally require iterative and joint training, leading to higher time costs.

D. PLURALISTIC INPAITING

Compared to certain inpainting problems, such as artifact inpainting and criminal identification, which require a single most accurate result, there exists a category of image inpainting problems characterized by open-ended and pluralistic inpainting outcomes. As long as the results are plausible, they are considered feasible solutions. Consequently, a series of

algorithms have been developed to generate multiple results by controlling latent representations. Based on the generation method of latent representations, pluralistic inpainting can be classified into methods based on GAN, VAE, Transformer, and Denoising Diffusion.

a: GAN-BASED PLURALISTIC INPAINTING

GANs [8] learn data distributions through adversarial training, where the generator converts Gaussian random noise into images and the discriminator distinguishes between real and generated samples. Diversity in inpainting based on GANs is achieved through the use of random vectors.

UCTGAN [54] is an unsupervised cross-space translation generative adversarial network that includes a conditional encoder, a manifold projection module, and a generation module. By projecting image spaces into a shared low-dimensional manifold space, it achieves unsupervised mapping, enhancing the diversity of the inpainted samples. The introduction of a cross-semantic attention layer leverages long-range dependencies, improving the realism and visual consistency of the inpainted samples. Based on StyleGAN, Karras et al. [118] optimized the mapping process from latent codes to images by normalizing the generator, improving the progressive generation process, and regularizing the generator. These enhancements aim to ensure high quality and stability in the generated results. PD-GAN [119] generates images from random noise. The generator infuses an initial restored image and multi-scale hole regions, repairing the input noise from coarse to fine levels. Following the principle that pixels near the boundary are more certain while those further away have higher degrees of freedom, the Spatial Probability Diversity Normalization (SPDNorm) method is proposed. This method simulates the probability of generating pixels based on context. Considering that the constraint of reconstruction loss hinders the diversity of the results, they also proposed a perceptual diversity loss. Zeng et al. [120] designed a model consisting of two consecutive steps: the first step formulates the inpainting process as a regression problem, using a U-Net-like convolutional neural network to map the input to a coarse inpainted output; the second step maps the coarse output to a high-quality output through pixel matching based on nearest neighbors. This step generates high-quality new content by copying and pasting high-frequency missing information from different training samples. Zhao et al. [121] proposed a method that builds a bridge between image-conditional generation and the latest unconditional modulation generation architectures by jointly modulating conditional and random representations. The cascaded Modulation GAN (CM-GAN) [122] consists of an encoder with Fourier convolution blocks and a dual-stream decoder. The encoder extracts multi-scale features of the image with holes, while the decoder employs cascade global-spatial modulation blocks at each scale level. The decoder blocks first synthesize coarse semantic structures through global modulation and then adaptively adjust the feature maps via spatial modulation. Yildirim et al. [123] developed a solution for inpainting and editing erased images by mapping them into the latent space of a GAN to achieve realistic results. They established an encoder and hybrid network that combines encoded features from the erased images with mapping features from random samples generated by StyleGAN.

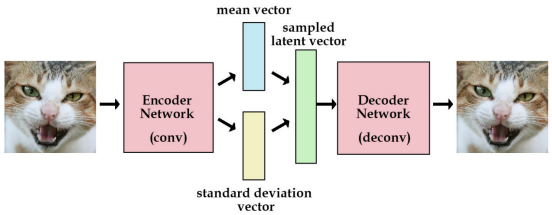

FIGURE 9. Variational Auto-Encoder [124].

b: VAE-BASED PLURALISTIC INPAINTING

An autoencoder(AE) [124] is a fundamental neural network architecture composed of an encoder and a decoder. The encoder compresses input data into a low-dimensional latent representation space, while the decoder reconstructs the input data from this low-dimensional space. The objective of an AE is to minimize the difference between the input data and the reconstructed data, known as the reconstruction loss. Although AEs excel in data compression and denoising, they have limitations in generating new data due to a lack of explicit modeling of the latent space. To overcome these limitations, the Variational Autoencoder (VAE) [124] was developed, shown in Figure 9 [124]. VAEs not only focus on reconstructing the input data but also introduce the concept of probabilistic modeling. By imposing a prior distribution (typically a Gaussian distribution) on the encoder output, VAEs can generate more diverse and realistic new data. The training objective of a VAE is to maximize the likelihood of the data while minimizing the KL divergence between the latent representation and the prior distribution.

Assume that $p_\theta(x \mid z)$ is either a multivariate Gaussian or Bernoulli distribution, computed from z using a multilayer perceptron (MLP), which makes $p_\theta(z \mid x)$ intractable. VAE utilize a neural network as encoder $q_\psi(z \mid s)$, serving as an approximation of $p_\theta(x, z)$. The encoder's outputs, $\mu^{(i)}$ and $\sigma^{2(i)}I$, which are nonlinear functions of $x^{(i)}$, define $q_\psi(z \mid x)$ as a Gaussian with diagonal covariance, expressed as

$$\log q_\psi(z \mid x^{(i)}) = \log \mathcal{N}(z; \mu^{(i)}, \sigma^{2(i)}I)$$

The parameters ψ and θ are optimized using the Auto-Encoding Variational Bayes algorithm, with the goal of maximizing the objective function.

$$L(\theta, \psi; x^{(i)}) = \frac{1}{2}\sum_{j=1}^{J}\left(1 + \log\left((\sigma_j^{(i)})^2\right) - (\mu_j^{(i)})^2 - (\sigma_j^{(i)})^2\right) + \frac{1}{L}\sum_{l=1}^{L}\log p_\theta\left(x^{(i)} \mid z^{(i,l)}\right)$$

Tu et al. [125] proposed a method that leverages a pre-trained standard Variational Autoencoder (VAE) to generate unoccluded facial images. Specifically, the approach involves searching the set of possible VAE encoding vectors for the given occluded input image to ensure the robustness of the predicted encodings for the missing regions. Subsequently, the decoder of the VAE model reconstructs a set of potential facial appearances from these encodings. FiNet [126] is a two-stage image generation framework that consists of a Shape Generation Network and an Appearance Generation Network. Each generation network incorporates two encoders that interact by learning latent codes in a shared, compatible space to enhance the quality of the generated results. Razavi et al. [127] extended and enhanced the autoregressive prior used in VQ-VAE by employing simple feedforward encoder and decoder networks. This improvement not only increased the consistency and fidelity of the generated samples but also improved the speed of encoding and decoding. Zheng [128] propose a framework comprising two parallel GANs. The first reconstruction path captures the prior distribution of the missing parts based on ground truth and reconstructs the image. The conditional prior of the second generation path is coupled with the distribution of the reconstruction path. The newly introduced short-term and long-term attention layer leverages long-range relationships between decoder and encoder features to enhance the visual consistency of the images. Zheng et al. [129] improved upon previous work by incorporating the prior distribution of missing patches into the reconstruction network to better restore the original image. They also introduced a new short-term and long-term patch attention layer, which leverages long-range relationships between decoder and encoder features to enhance the visual consistency between the original visible parts of the image and the newly generated regions. Peng et al. [130] proposed a two-stage image inpainting model. In the first stage, multiple coarse inpainting results are generated, and in the second stage, details are enhanced for fine inpainting. The model draws on the hierarchical VQ-VAE structure, separating the processing of structural and textural information in the image, thereby improving the diversity and quality of the inpainting results. Liu et al. [131] proposed the "PUT" framework based on Transformers, called Patch VQ-VAE (PVQVAE), where the encoder transforms masked images into non-overlapping patch tokens, and the decoder restores the masked regions while preserving the unmasked regions unchanged. To eliminate quantization losses, they employed a non-quantized transformer, UQ-Transformer, which directly utilizes features from the PVQVAE encoder, with quantized tokens used solely as prediction targets.

c: TRANSFORMERS-BASED PLURALISTIC INPAINTING

Transformers, due to their inherent properties, do not require any priors and can directly perform maximum likelihood optimization to generate output distributions. By sampling from these distributions, diverse results can be obtained, making them highly suitable for multi-modal image inpainting.

BAT-Fill [132], an image inpainting framework that utilizes a Bidirectional Autoregressive Transformer (BAT) to learn autoregressive distributions and integrates a BERT-like masked language model for bidirectional context modeling, thereby enhancing the inpainting performance. Wan et al. [133] leveraged Transformers to restore multi-modal coherent structures and some coarse textures, while employing Convolutional Neural Networks (CNNs) to enhance the local texture details of the coarse priors guided by high-resolution masked images. Liu et al. [134] designed an image inpainting network based on the Swin Transformer, utilizing the shifted window scheme for representation computation. This shifted window approach confines the self-attention computation to non-overlapping local windows while allowing cross-window connections, thereby enhancing efficiency. MAT [135] is based on the Transformer framework and features a Multi-head Contextual Attention (MAT) component, which efficiently models long-range dependencies using dynamically masked valid tokens. The improved Transformer model structure enhances stability when training on large masked images. Additionally, a novel style manipulation module is designed to provide pluralistic image inpainting results. Campana et al. [136] proposed a variable hyperparameter vision transformer architecture with three key innovations: first, the feature maps are divided into a variable number of multi-scale patches; second, the feature maps are allocated to a variable number of heads to balance the complexity of self-attention operations; third, a new strategy based on deep convolution is included to reduce the number of feature map channels sent to each transformer block. The Bidirectional [137] Interactive Dual-Stream Network (BIDS-Net) combines the strengths of CNNs and Transformers. The CNN stream captures local patterns for detail reconstruction, while the Transformer stream models long-range context to leverage global information. A Bidirectional Feature Interaction (BFI) module for selective feature fusion is designed to enhance the locality of the Transformer and the global awareness of the CNN. The hierarchical encoder-decoder structure promotes multi-scale contextual reasoning and improves computational efficiency.

d: DENOISING DIFFUSION PLURALISTIC INPAINTING

The core idea of denoising diffusion models is derived from the diffusion process in physics, which describes the movement of particles from regions of high concentration to low concentration. In deep learning, this process is applied in reverse for data generation. Initially, during the forward process, noise is progressively added to the data samples, making them increasingly blurred until they become pure noise. This process is typically modeled using a Markov chain, starting from the data distribution X_0 and gradually adding Gaussian noise to generate a series of intermediate states approaching a standard Gaussian distribution.

The subsequent reverse process, which is the inversion of the forward process, involves progressively denoising from pure noise to generate the target data samples. The key to training the reverse process is learning how to recover the original data distribution step by step from the noise. By training a neural network to estimate the noise reduction at each step, it becomes possible to restore the original data from the noisy input.

Ho et al. [26] designed a Denoising Diffusion Probabilistic Models (DDPM) based on the new connection between diffusion probabilistic models and denoising score matching with Langevin dynamics. This model is trained using a weighted variational bound, naturally adopting a progressive lossy decompression scheme, which can be interpreted as a generalization of autoregressive decoding. RePaint [138] is a novel image inpainting technique specifically designed to address the challenge of extreme mask inpainting. This method leverages DDPMs, using a pre-trained unconditional DDPM model as a generative prior. By sampling the unmasked regions based on the given image information and adaptively adjusting during the reverse diffusion process, it produces high-quality and pluralistic inpainting results. A significant feature of RePaint is that it requires no modifications to the DDPM network structure, making it suitable for various image inpainting tasks. SmartBrush [139] is a method that combines text and shape guidance to achieve precise control in image generation and inpainting. This approach introduces a novel training and sampling strategy, enhancing the diffusion U-net through object mask prediction to better preserve background information. Additionally, SmartBrush incorporates a multi-task training strategy, jointly training on image inpainting and text-to-image generation tasks. This leverages a larger set of training data, thereby improving model performance and generation quality. ImageBART [140] is an image generation method that combines autoregressive formulation with polynomial diffusion processes. The model progressively compresses and removes information to coarsen the image through a multi-stage diffusion process, while training a Markov chain to reverse this process. At each stage, the ImageBART model gradually integrates contextual information from the previous stage in an autoregressive manner, thereby achieving image generation from coarse to fine details. Hoogeboom [141] introduced two extensions for handling categorical data: Argmax Flow and Polynomial Diffusion. Argmax Flow combines continuous distributions with the argmax function by learning the probability inverse of the argmax function to map categorical data to a continuous space for optimization. Polynomial Diffusion, on the other hand, incrementally introduces categorical noise during the diffusion process and learns the denoising process to gradually recover the original data. Austin et al. [142] introduced the Discrete Denoising Diffusion Probabilistic Models (D3PMs), a diffusion generative model for discrete data that improves upon Hoogeboom et al.'s polynomial diffusion model. These improvements include simulating Gaussian kernel transition matrices in continuous space, utilizing neighborhood-based matrices in embedding space, and introducing a matrix corruption process with absorbing states. Rombach et al. [143] designed a model that applies Diffusion Models (DM) within the latent space of a powerful pre-trained autoencoder, achieving an optimal balance between complexity reduction and detail preservation, thereby significantly enhancing visual fidelity. By introducing cross-attention layers into the model architecture, the diffusion model is transformed into a robust and flexible generator capable of handling general conditional inputs, such as text or bounding boxes, and achieving high-resolution synthesis in a convolutional manner. The Come-Closer-DiffuseFaster [144] model leverages the contraction theory of stochastic differential equations to demonstrate that starting from Gaussian noise is unnecessary. Instead, beginning with a single forward diffusion with better initialization can significantly reduce the sampling steps required for the reverse conditional diffusion. Additionally, this model provides new insights into how existing feedforward neural network methods can be synergistically combined with diffusion models to solve inverse problems. Peter et al. [145] proposed a neural network architecture designed for the rapid optimization of both pixel positions and pixel values, achieving uniform diffusion coloring. During the training process, this architecture integrates two optimization networks with a neural network-based diffusion coloring proxy solver. Generative Diffusion Prior (GDP) [146] explores a conditional guidance protocol that leverages a pre-trained Denoising Diffusion Probabilistic Model (DDPM) to address linear inverse, nonlinear, or blind problems through an unsupervised sampling approach to effectively model the posterior distribution. The method optimizes the degradation model parameters during the denoising process for blind image inpainting. Additionally, the model's design incorporates hierarchical guidance and patch-based methods, enabling GDP to generate images of arbitrary resolution.

e: SUMMARY OF GENERATOR

Multi-networks determine whether the network generator comprises multiple networks and how these networks are arranged. Connections define the structure both within and between modules in a network, and layers constitute the fundamental building blocks of the network. Together, these three components form a complete generator. Therefore, classifying strategies according to these three categories not only comprehensively covers existing and future connection-based strategies but also provides a clear direction for future research to explore these strategies more explicitly. The ultimate goal of multi-strategy inpainting is to create new strategies aimed at different objectives. The fundamental principle of these strategies is to enable the model to fit the pixel and feature distributions of the image as closely as possible, and subsequently select multiple high-probability outcomes from these distributions. In this process, we find

that controlling such distributions can also allow for a certain degree of image and feature modification and editing.

III. LOSS FUNCTIONS AND EVALUATION METRICS

The loss function is central to optimization algorithms, as it adjusts model parameters by minimizing the loss function to better fit the training data. Applied to the training set, the loss function is crucial for the training speed and effectiveness of the model, as different loss functions can lead to variations in the rate of gradient descent. Evaluation metrics assess the model's performance on the test set and are typically used to measure the model's generalization ability and practical effectiveness. There are many algorithmic similarities between image inpainting evaluation metrics and these loss functions. Both are used to assess image quality; however, they are employed at different stages, and the loss function must be differentiable to guide model optimization and parameter updates.

Moreover, evaluation metrics typically compare the generated images with ground truth images, but loss functions do not always follow this approach. Most current image inpainting models employ supervised learning, where paired corrupted images and their corresponding ground truth images are used for training. Within this supervised learning framework, loss functions, apart from the Total Variation (TV) loss, usually compare the generated images to the ground truth images. However, in certain domains where image data is scarce or difficult to obtain, zero-shot learning, unsupervised learning, and weakly supervised learning have increasingly emerged. Unsupervised models, in particular, train solely on corrupted images without the corresponding original ground truth images. These models, therefore, rely on surrogate tasks to generate self-supervised signals, which serve as target restoration images. The loss functions then compute the discrepancy between these target images and the generated images, guiding the optimization process.

Zero-shot image inpainting, on the other hand, often depends on deep learning models pre-trained on large-scale datasets. By extracting general features or knowledge from these pre-trained models, zero-shot image inpainting can transfer these features to the target images. The model leverages this learned knowledge to infer plausible content for the missing regions of the image. To ensure that the restored content is consistent with the unmasked areas of the original image, zero-shot image inpainting models may incorporate consistency loss functions. These constraints help generate restoration results that are harmonized with the surrounding areas, thereby maintaining contextual coherence.

Overall, the target objects for comparison in loss functions, whether they involve the original ground truth image, latent space representations or features, or the intact regions of a corrupted image, can be classified into three categories, along with the corresponding evaluation metrics: pixel-based algorithms, feature-based algorithms, and model-based algorithms, as shown in Figure 10.Classification of loss functions and evaluation metrics. The following section provides a detailed introduction to these algorithms.

A. PIXEL-BASED

An image is composed of a series of pixels, making pixel-based loss measurements and evaluation metrics the earliest and most fundamental methods for assessing inpainting quality. Pixel-based algorithms primarily include the basic algorithms based on the first norm (L1 norm) and the second norm (L2 norm), as well as their improved versions. These algorithms rely on a set of mathematical formulas, offering fast computation but primarily focusing on pixel-level evaluation, thus lacking in higher-level semantics.

1) LOSS FUNCTION

Pixel-based algorithms form the foundation of loss functions and evaluation, providing relatively precise guidance for model optimization. Some models refer to these as reconstruction loss. The basic pixel-based algorithms include L1 Loss and L2 Loss. Improved loss functions derived from these include Hole and Valid Reconstruction Loss, Weighted Reconstruction Loss, Multi-scale Reconstruction Loss, and Total Variational Loss.

a: BASIC PIXEL-BASED LOSS

L1 and L2 Loss: Also known as pixel-wise loss or reconstruction loss is the most fundamental type of loss used in image inpainting. The L1 loss function is the sum of the absolute values of the errors. It contributes linearly to the total loss, meaning each error, regardless of its size, contributes equally to the total loss. The formula for the L1 loss function is as follows:

$$\mathcal{L}_1 = \sum_{i=1}^{N} ||I^i_{generate} - I^i_{target}||_1$$

The L2 loss function is the sum of the squared errors. This means that large errors contribute much more to the total loss than small errors.

$$\mathcal{L}_2 = \sum_{i=1}^{N} ||I^i_{generate} - I^i_{target}||_2$$

where I^i_{target} represents the true value of the missing region for the i-th sample, $I^i_{generate}$ denotes the predicted value of the missing region for the same i-th sample, and N denotes the total number of samples.

The linear relationship in the L1 loss function makes it less sensitive to outliers or large errors compared to the L2 loss function. This is because L1 loss treats large errors with the same weight as small errors when calculating the total loss. In the optimization process, L1 loss encourages the model to generate fewer large errors, as these errors do not have an excessive impact on the loss function compared to L2 loss. In image reconstruction tasks, L1 loss helps to preserve edges and details because it does not excessively penalize large

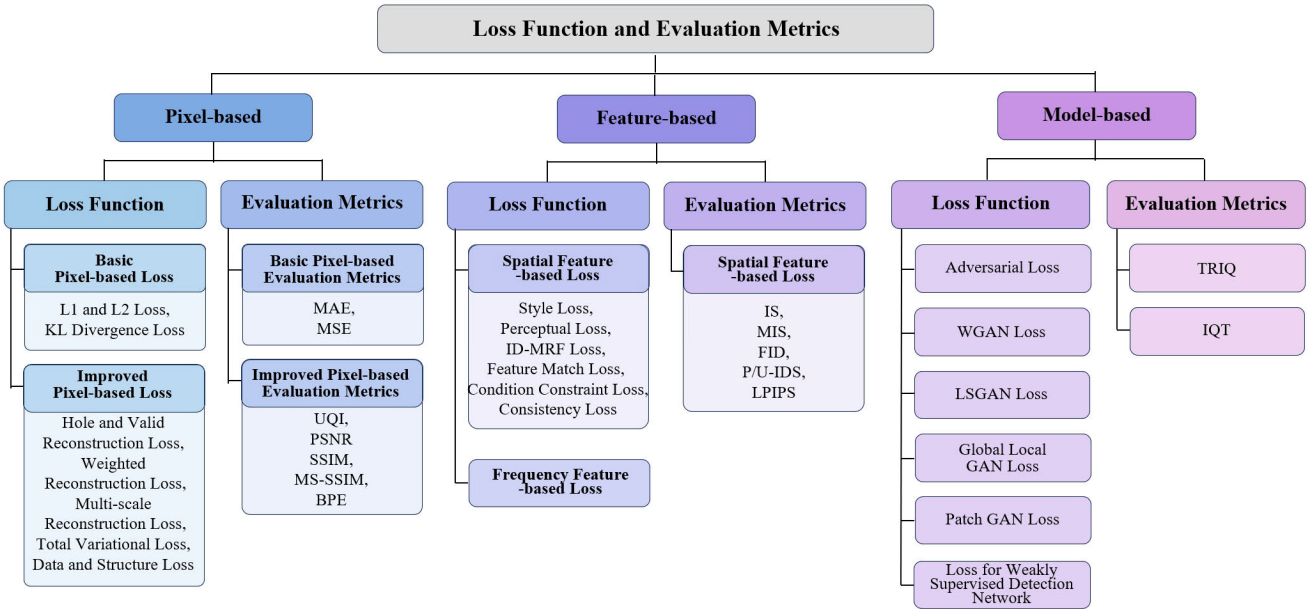

FIGURE 10. Classification of loss functions and evaluation metrics.

pixel differences, which might be important features in the image. Minimizing L2 loss, on the other hand, leads the model to pay particular attention to reducing large errors, as these errors contribute significantly to the total loss due to their squared nature. This high sensitivity to large errors causes the optimization process to avoid significant pixel intensity changes, which introduces a smoothing effect in the reconstructed image. In image reconstruction, this smoothing effect can lead to blurring, as the model tends to predict averaged results to achieve a better overall score.

KL Divergence Loss: KL divergence measures the distance between two probability distributions and is commonly used in unsupervised learning, particularly in variational autoencoders, to constrain the distribution of latent variables. The formula for KL divergence is given by:

$$D_{KL}(P \parallel Q) = \sum_{x} P(x) \log \frac{P(x)}{Q(x)}$$

where $P(x)$ and $Q(x)$ represent the true probability distribution and the model's output probability distribution, respectively, and x denotes elements in the sample space.

In practical applications of machine learning, the true probability distribution $P(x)$ is often unknown, so we typically approximate $P(x)$ using the empirical distribution derived from the training data. This approximation involves estimating $P(x)$ as the frequency of each sample's occurrence in the training set. Meanwhile, $Q(x)$ is the probability distribution output by the model given the input data.

The smaller the KL divergence loss, the closer the model's output distribution is to the true data distribution, indicating better model performance. Therefore, during training, we typically minimize the KL divergence loss to optimize the model parameters.

b: IMPROVED RECONSTRUCTION LOSS

L1 Loss and L2 Loss emphasize the prediction of each pixel vector, assuming that each pixel in the image has the same learning capacity. This characteristic makes pixel-based losses sensitive to outliers or noise at the individual pixel level, but they fail to capture higher-level feature information of the image, thereby affecting overall predictive performance. To address these shortcomings, a series of improved reconstruction losses have been developed to enhance structural and semantic optimization.

Hole and Valid Reconstruction Loss: Generally, the reconstruction loss is calculated for the defective region being restored. However, in Partial Convolution [20], to evaluate and improve the model's generative capability, not only is the L1 loss used to enhance the generation of the defective parts named hole loss, but constraints are also applied to the non-defective parts named valid loss. This approach ensures that the generated regions are more coherent.

Weighted Reconstruction Loss: The fundamental idea behind weighted reconstruction loss is that areas closer to the edges of a hole are easier to repair and more certain, hence should have higher weights, whereas the central parts of the hole are more uncertain and should have lower weights. Based on this concept, Yeh et al. [71] proposed a semantic-guided image inpainting model using a weighted reconstruction loss function to optimize the model. Similarly, the GA model [15] introduced Spatially Discounted Reconstruction Loss, which uses a weight mask to reconstruct the loss, drawing inspiration from reinforcement learning, where

temporal discounting is applied to sampling trajectories when high variance is observed in long-term rewards [147]. Likewise, Lahiri et al. [112] proposed the Spatially Adaptive Contextual Loss, which determines the weight by using 0.99 as the base and the distance to the boundary as the exponent. Wang et al. [14] proposed the Spatial Variant Reconstruction Loss, which defines a loss weight mask matrix where known pixels are set to 1, and the confidence in the unknown regions is obtained by convolving with a Gaussian filter.

Multi-scale Reconstruction Loss: Multi-scale reconstruction loss leverages the rich semantic information contained in multiple layers and scales within a network to better guide model optimization. Research has shown that lower layers and upper layers are expected to learn low-level and high-level information, respectively. Therefore, even in columnar architectures with feature maps of the same size at all layers, using single-scale guidance for multiple local layers is not appropriate. Multi-scale reconstruction loss provides fine-scale supervision for lower layers and coarse-scale supervision for higher layers [148]. The PEN [21] outputs each layer's features as an RGB image through a 1×1 convolution operation and resizes the ground truth image into multi-scale thumbnails matching the size of each U-Net layer. The L1 loss between these generated images and the corresponding resized ground truth images is computed, referred to as Pyramid L1 losses. Yang et al. [88] proposes a pyramid structure loss to guide the generation and embedding of structures, thus incorporating structural information into the generation process. Specifically, it consists of two items. One is the L1 distance between the predicted gradient map and the corresponding ground truth. The other is the regularization term used to learn the edge structure. To achieve regularization on the edge structure, the binary ground truth edge map is first convolved using a Gaussian filter, and then a weighted edge mask is created, which imposes constraints on the locations in its vicinity, thus highlighting and strengthening the edge structure.

Total variation (TV) Loss: The vast majority of loss functions are employed to quantify the discrepancy between model predictions and actual values. However, TV loss [149], which does not require a reference image, is commonly used for smoothing in image denoising tasks. In the context of image inpainting, TV loss is employed to constrain noise and support other losses. The total variation operation reduces the overall variation of the signal, making it more akin to the original signal to achieve image smoothing. Additionally, it preserves significant details, such as edges, while removing insignificant ones. The formula of TV Loss is as following:

$$L_{TV}(x) = \sum_{i,j} \frac{||x_{i+1,j} - x_{i,j}||_1}{N} + \sum_{i,j} \frac{||x_{i,j+1} - x_{i,j}||_1}{N}$$

where i and j are the horizontal and vertical coordinates of pixels in the image, N is the total number of pixels in the image, and x_{ij} denotes the pixel value at position (i,j).

Data and Structure Loss: In unsupervised learning, [150] introduced Data Loss and Structure Loss, both of which use L1 Loss to measure the similarity between the generated image and the undamaged portions of the corrupted image. If the discrepancy between the generated and actual undamaged regions is significant, it suggests that the inferred missing parts are also unreliable, thereby incurring a penalty. The difference lies in the basis of the losses: Data Loss is pixel-based, while Structural Loss is derived from the horizontal and vertical gradient values calculated between neighboring pixels.

2) EVALUATION METRICS

Pixel-based evaluation metrics can quickly assess the differences between the inpainting target and the inpainting result, making them essential evaluation criteria in single-instance inpainting models. However, in multi-instance inpainting models, where the relationship between the inpainting target and the inpainting result is one-to-many, pixel-based evaluations become less effective.

a: BASIC PIXEL-BASED EVALUATION METRICS

Mean Absolute Error (MAE): MAE [151] refers to the average value of the total absolute difference between the pixel values of the repaired image and the original image. It is mainly used to evaluate the difference between the repaired image and the original image and lower of the value means closer to the ground truth. The calculation formula is shown following:

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |x_i - y_i|$$

The 'x' represents the generated image, the 'y' represents the original image, the 'N' represents the total number of image pixels, and the 'i' represents the variable of image pixel. When comparing images, if all images N are considered simultaneously, there are instances where division by N is not applied. Thus, the essence of MAE and L1 Loss remains the same.

Mean Squared Error (MSE): MSE [152] refers to the average value of the sum of the squares of the difference between the pixel values of the repaired image and the original image, which is mainly used to evaluate the similarity between the repaired image and the original image and lower of the value means closer to the ground truth. The calculation formula is shown following:

$$MSE = \frac{1}{N} \sum_{i=1}^{N} \|x_i - y_i\|^2$$

The 'x' represents the generated image, the 'y' represents the original image, the 'N' represents the total number of image pixels, and the 'i' represents the variable of image pixel. When comparing images, if all images N are considered simultaneously, there are instances where division by N is not

applied. Thus, the essence of MSE and L2 Loss remains the same.

b: INPROVED PIXEL-BASED EVALUATION METRICS

Peak Signal to Noise Ratio (PSNR): PSNR [153] is an important indicator to measure image quality. In the process of image inpainting, the higher the PSNR value of the inpainting result shows that the restored image is closer to the ground truth. The PSNR formula is shown following:

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX^2}{MSE}\right)$$

MAX represents the maximum value of pixel signal in the generated image, and MSE represents the MSE between the generated image and the original image.

Structural Similarity Index Measurement (SSIM): SSIM [154] evaluates the similarity of two images by measuring the Luminance, Contrast, and Structural of the generated image and the original image and higher of the value means closer to the ground truth. The Luminance of images 'x' is measured by the average gray scale of each channel, which is obtained by averaging the values of all pixels, The formula is shown following:

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i$$

The Luminance similarity function of the generated image 'x' and the original image 'y' is shown following:

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}$$

The Contrast is measured by the standard deviation of gray scale of each channel in image 'x'. Unbiased estimation of standard deviation is shown following:

$$\sigma_x = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2}$$

he Contrast similarity function of the generated image 'x' and the original image 'y' is shown following:

$$c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}$$

The Structure function is related to the correlation coefficient of 'x' and 'y' shown following:

$$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y)$$

The Structure similarity function of the generated image 'x' and the original image 'y' is shown following:

$$s(x, y) = \frac{2\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}$$

The SSIM equation of the generated image 'x' and the original image 'y' is shown following:

$$\mathrm{SSIM}(x, y) = \left[l(x, y)^\alpha \cdot c(x, y)^\beta \cdot s(x, y)^\gamma\right]$$

α, β, γ respectively represent the proportions of different characteristics in SSIM measurement. When they equal 1, the SSIM is shown following:

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

Multi-Scale Structural Similarity Index (MS-SSIM): MS-SSIM [155] is an advanced version of the SSIM. It processes two input images iteratively through a low-pass filter and performs four downsampling operations with a stride of 2, resulting in five images of different scales. Contrast and structure comparisons are computed at each scale, while luminance comparison is conducted at the final scale.

Border Pixel Error (BPE): BPE [19] evaluates the quality of boundary inpainting in generated images by calculating the pixel error near the boundary region. The calculation formula is shown in Equation:

$$\mathrm{BPE} = \frac{\|b \odot (I - \hat{I})\|_1}{\|b\|_1}$$

where b represents the boundary region of the image, I denotes the original image, and $\hat{I}$ denotes the generated image.

B. FEATURE-BASED

Feature-based loss and evaluation algorithms are better suited for assessing and optimizing a model's ability to learn textures, structures, and even high-level semantic information. Compared to pixel-based algorithms, feature-based algorithms more closely align with human cognition. However, these algorithms require the use of feature maps. While some feature maps, such as those based on frequency characteristics, can be directly obtained through mathematical algorithms, most other feature maps are derived from pre-trained models, such as convolutional models based on VGG16/19 [6], ResNet [7], and Inception [28], [29], [30]. This reliance is due to convolutional neural networks (CNNs) having rotational and translational invariance, allowing them to effectively extract and represent perceptual feature information. However, this also means that feature-based algorithms are, to some extent, dependent on the structure, training data, and training processes of these pre-trained models, which limits their absolute effectiveness.

1) LOSS FUNCTION

Compared to pixel-based loss functions, feature-based loss functions optimize image inpainting models using higher-level features, ignoring the influence of individual pixels. During model training, these two types of loss functions often complement each other, resulting in better inpainting outcomes. Feature-based loss functions can be divided into spatial-based losses and frequency-domain-based losses.

a: SPATIAL FEATURE-BASED LOSS

Widely used spatial feature-based losses include perceptual loss, style loss, and Markov Random Field (MRF) loss. Perceptual loss, originating from style transfer tasks, evaluates the difference between generated images and undamaged images at the feature level. Style loss focuses on capturing the texture differences between images. MRF loss is typically used for image segmentation and fine texture generation to achieve higher detail recovery.

Perceptual loss: Perceptual loss, also known as content loss in style transfer, primarily evaluates image quality from the perspective of content understanding and perception. It typically uses a pre-trained image classification network to assess the semantic differences between two images, thereby enhancing the visual similarity and realism of the reconstructed image. The basic formula for perceptual loss is given as follows:

$$\mathcal{L}_{\text{perceptual}} = \frac{1}{L}\sum_{i=1}^{N}\sum_{l=1}^{L}\left\|\phi_l(\mathbf{I}^i_{generated}) - \phi_l(\mathbf{I}^i_{target})\right\|_1$$

where ϕ_l represents the feature map obtained from the pre-trained convolutional neural network at layer l, $\mathbf{I}^i_{generated}$ is the generated image for the i-th sample, $\mathbf{I}^i_{target}$ is the target image for the i-th sample, $\mathbf{L}$ is the number of feature layers, and $\mathbf{N}$ is the number of samples. However, in practical applications of image inpainting models, due to the fixed number of feature extraction layers, many studies omit the operation of dividing by L layers from the formula or replace it with the product of channel and feature size. Based on this formula, they made improvements for specific models. Partial convolution [20] divides perceptual loss into two parts: one part is the perceptual loss between the actual generated image and the ground truth image, and the other part directly compares the result of setting non porous pixels as ground truth with the ground truth image. The WaveFill [156] model specifically weights the relu4_2 layer in VGG-19, treating it as a special layer to enhance the model's advanced semantic understanding.

Style loss: Style loss primarily originates from the field of style transfer, where it is commonly referred to as texture loss. It typically measures the similarity between the reconstructed and original images in terms of color, texture, and contrast by evaluating the correlations between different channels of the two images. Different channels in a feature map record different characteristics of an image. To amplify these characteristics, the Gram matrix between feature maps is computed. The Gram matrix is a matrix that measures the features shown following.

$$G^{\phi}_l(x)_{c,c'} = \frac{1}{C_l H_l W_l}\sum_{h=1}^{H_l}\sum_{w=1}^{W_l}\phi_l(x)_{h,w,c}\phi_l(x)_{h,w,c'}$$

where ϕ_l represents the feature map obtained from the pre-trained convolutional neural network at layer l and $\mathbf{C_l} \times \mathbf{H_l} \times \mathbf{W_l}$ is the size of $\phi_l(x)$ at layer l. Style loss is based on the first paradigm of the Gram matrix, with the specific formula given as follows:

$$\mathcal{L}_{\text{style}} = \sum_{i=1}^{N}\sum_{l=1}^{L}\left\|G^{\phi}_l(\mathbf{I}^i_{generated}) - G^{\phi}_l(\mathbf{I}^i_{target})\right\|_1$$

where G^{ϕ}_l represents the Gram matrix obtained from the pre-trained convolutional neural network ϕ_l at layer l, $\mathbf{I}^i_{generated}$ is the generated image for the i-th sample, $\mathbf{I}^i_{target}$ is the target image for the i-th sample, $\mathbf{L}$ is the number of feature layers, and $\mathbf{N}$ is the number of samples. Similar with perceptual loss, Partial convolution [20] divides style loss into global perceptual loss and progressive loss.

ID-MRF Loss: This image exemplifies a typical Markov Random Field (MRF), where each point may be related to its surrounding points but has minimal connection to distant or initial points. The influence of points increases with proximity. Unlike L1 or L2 distance measures, MRF loss evaluates differences between high-level image representations. For each patch in the missing area, MRF loss calculates its distance to the nearest neighbor in the known area. Yang et al. [82] employed Euclidean distance to identify the nearest neighbors of each patch. However, this method often results in smooth structures, making patches in missing areas look similar. GMCNN [14] used the relative distance to instead of Euclidean distance. A patch in a missing area will have a larger relative distance if it is close to a patch in a known area and a shorter relative distance if it is close to a different patch in a known area.

Feature Match Loss: The definition of feature match loss is similar to perceptual loss, but instead of using features from an external VGG model, it directly utilizes the activation values from each layer of the discriminator. The primary advantage of this approach is that the VGG model, trained for classification, lacks edge information, whereas the discriminator is not constrained in this way. Consequently, a series of models have adopted feature match loss. Models such as EdgeConnect [16], pix2pix [157], and WaveFill [156] have incorporated Feature Match Loss into their optimization functions to generate more plausible textures.

Condition Constraint Loss: In unsupervised learning, Condition Constraint Loss [54] comprehensively calculates the pixel-level and feature-level losses between the corrupted image and the repaired image. The pixel-level loss is computed using a simple pixel-based L1 loss, referred to as the appearance constraint loss. The feature-level loss employs a pre-trained VGG feature extractor to extract image features, similar to perceptual loss, and is referred to as the perceptual constraint loss.

Consistency Loss: Consistency losses primarily include content consistency loss and texture consistency loss. Both are mainly used in zero-shot learning and are designed to ensure that the generated inpainting regions align with the unmasked parts of the image. Content consistency loss is similar to perceptual loss, as it is based on features extracted from a pre-trained VGG network. Texture consistency loss,

on the other hand, is akin to style loss, as it is computed using the Gram matrix of the texture. However, in some cases, both types of losses measure the L2 distance between the generated and intact regions.

b: FREQUENCY FEATURE-BASED LOSS

Perceptual loss is primarily designed for the spatial domain. However, during downsampling, super-resolution predominantly loses high-frequency information in the spectral domain. To address this, Jiang et al. [158] proposed a frequency-domain-based loss function called Focal Frequency Loss (FFL). The core idea of FFL is to selectively adjust local and global image information by focusing on different frequency components. Firstly, each spectral coordinate value is mapped to a Euclidean vector in two-dimensional space, considering both the amplitude and phase information of the spatial frequencies. Secondly, the Euclidean distances of these vectors are scaled. Thirdly, simple frequencies are down-weighted using a dynamic normalization weight matrix. Lastly, the model rapidly focuses on hard frequencies and gradually refines the generated frequencies to enhance image quality. This approach can produce images with rich detail variations, helping deep learning models better understand and learn the patterns within images.

2) EVALUATION METRICS

Feature-based image inpainting evaluation metrics are mostly used in multi-instance inpainting. This is because single-instance inpainting emphasizes absolute similarity to the target function, whereas multi-instance inpainting focuses more on generating visually continuous and plausible results rather than exact replicas of the original image. Feature-based evaluation metrics can better satisfy the assessment of high-level semantic and structural similarities.

a: SPATIAL-BASED EVALUATION METRICS

Inception Score (IS): IS [159] utilized the Inception-v3 [29] network, initially proposed by Google in 2014 for image classification on the ImageNet dataset. This network takes an image as input and outputs a 1000-dimensional tensor representing the output classes.

Let x be a generated image, and let y be the classification result from the discriminator for this generated image. In IS, the classification involves 1000 categories. The higher the quality of the image, the more confident and stable the discriminator's classification result will be (i.e., the probability of belonging to a specific class will be higher), leading to a lower entropy of $P(y|x)$. The specific formula for this is:

$$\mathrm{IS} = \exp\left(\mathbb{E}_x\left[D_{\mathrm{KL}}(P(y|x) \parallel P(y))\right]\right)$$

where D_{KL} represents the Kullback-Leibler divergence between the conditional probability $P(y|x)$ and the marginal probability $P(y)$.

IS can reflect both the diversity and quality of the data. It is suitable for situations where the classification model and the generated model's dataset are similar. However, it has some limitations, such as its sensitivity to the internal weights and its inability to distinguish overfitting. Although IS is frequently cited in academic papers, its stability is not sufficient for practical use in production models where the dataset continuously iterates. Therefore, this metric may not be reliable for evaluating the iterations of the model itself.

Modified Inception Score (MIS): MIS [54] is an evaluation metric proposed by Zhao et al. to improve the original IS [159] for better assessing the quality of generated and restored images. Compared to IS, MIS is more suitable for evaluating the quality of images in image inpainting tasks. The calculation formula is shown following:

$$\mathrm{MIS} = \exp\left(\mathbb{E}_{x\sim p_g}\left[\sum_i p(y_i|x)\log_{10} p(y_i|x)\right]\right)$$

where x denotes the original image, y represents the labels predicted by a pre-trained model, p_g is the model distribution of real images, and i is the index over the generated images.

Frechet Inception Distance (FID): FID [160] is one of the evaluation metrics used for pluralistic inpainting tasks, initially proposed in 2017. Due to the limitations of IS on ImageNet, when the generated data samples fall outside the range of ImageNet, the performance of IS deteriorates. Consequently, FID measures the distance between the distribution of generated data and real-world data. Similar to IS, FID also utilizes the Inception-v3 model. However, FID does not directly use the classification results from Inception-v3; instead, it extracts image features from the final pooling layer. By calculating the means and covariances of the restored and target images, the outputs of the activation functions are summarized into a multivariate Gaussian distribution. These statistics are then used to compute the Frechet distance between the real and generated image sets. Since the Frechet distance focuses on the movement of one distribution to another in a multidimensional space, it exhibits good generalization capabilities even for images that differ significantly from those in ImageNet.

$$\mathrm{FID} = \left\|\mu_r - \mu_g\right\|^2 + \mathrm{Tr}\left(\Sigma_r + \Sigma_g - 2\left(\Sigma_r\Sigma_g\right)^{\frac{1}{2}}\right)$$

where μ_r and μ_g are the means, and Σ_r and Σ_g are the covariances of the real and generated data distributions, respectively. Frechet Inception Distance (FID), like Inception Score (IS), also relies on the presence or absence of existing features and focuses on textures. However, it cannot detect some abnormal structures that may appear in generated images and is unable to distinguish overfitting.

Kernel Inception Distance (KID): KID [161] does not assume a normal distribution like FID and is an unbiased estimate. Unlike FID, KID uses the maximum mean discrepancy (MMD) method to calculate the values of the 2048-dimensional Inception features of the images in the mapping

space for two different distributions. It measures the distance between the two distributions by comparing the generated samples with the real samples to evaluate the effectiveness of image generation.

$$\mathrm{KID} = \left\| \frac{1}{m}\sum_{i=1}^{m}\phi(x_i) - \frac{1}{n}\sum_{j=1}^{n}\phi(y_j) \right\|^2$$

where $\phi(x)$ and $\phi(y)$ are the mappings of the Inception features of the real and generated samples, respectively. Compared to FID, KID is unbiased, while FID is biased. In terms of computational efficiency, FID operates in $O(n)$ time complexity, whereas KID operates in $O(n^2)$ time complexity.

Paired/Unpaired Inception Discriminative Score (P-IDS/U-IDS): Inspired by the human assessment of image inpainting effects, Zhao et al. proposed the Paired/Unpaired Inception Discriminative Score (P-IDS/U-IDS) [121]. This method measures linear separability in the 2048-dimensional feature space of the pre-trained Inception v3 model, followed by fitting with a linear SVM. P-IDS is utilized for comparing paired data, whereas U-IDS is employed in the absence of paired information. Their primary advantages over the Fréchet Inception Distance (FID) include robustness to sample size, effectiveness in capturing subtle differences, and strong correlation with human preferences.

Learned Perceptual Image Patch Similarity (LPIPS): LPIPS [162] is a learned perceptual similarity metric. Compared to PSNR [134] and SSIM [135], it better aligns with human perceptual capabilities. The calculation formula is shown following:

$$\mathrm{LPIPS}(x, y) = \sum_{l}\frac{1}{H_l W_l}\sum_{h=1}^{H_l}\sum_{w=1}^{W_l}\|\phi_l(x)_{hw} - \phi_l(y)_{hw}\|_2^2$$

where ϕ_l denotes the activations from layer l of a pre-trained deep network, and H_l and W_l are the height and width of the l-th layer activations, respectively. Compared to FID, LPIPS also utilizes internal activations of deep convolutional networks. However, LPIPS measures perceptual similarity rather than quality assessment.

C. MODEL-BASED

Model-based loss primarily refers to the adversarial loss in GANs and their various improved versions. Inspired by this, some studies have started using deep learning models to evaluate the quality of image generation. Although these applications are still relatively few, they represent a promising direction for the optimization of future evaluation metrics.

1) LOSS FUNCTION

a: GAN LOSS

Following the introduction of the GAN model by Goodfellow et al. [8], which is also known as valina GAN, in 2016, they applied the GAN model to image generation [163]. The generator first captures the distribution of random noise within the image and then generates samples similar to real data based on this noise distribution to serve as input for the discriminator. The discriminator's function is to estimate the probability that a given sample comes from the generator, determining whether the input data is a real image or a generated sample. During the training process, the generator continuously improves its ability to generate sufficiently realistic samples to deceive the discriminator, while the discriminator enhances its ability to discern the authenticity of the samples. Through this adversarial learning process between the generator and discriminator, the model is optimized. Ultimately, the two networks reach a dynamic equilibrium. The GAN loss function is formulated as follows:

The Generative Adversarial Network (GAN) loss functions for the generator and discriminator are defined as follows:

Discriminator Loss:

$$\mathcal{L}_D = -\mathbb{E}_{\mathbf{x}\sim p_{\mathrm{data}}(\mathbf{x})}[\log D(\mathbf{x})] - \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))]$$

Generator Loss:

$$\mathcal{L}_G = -\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log D(G(\mathbf{z}))]$$

Here, $D(\mathbf{x})$ represents the discriminator's probability estimate that the real data instance $\mathbf{x}$ is real. $D(G(\mathbf{z}))$ represents the discriminator's probability estimate that a fake data instance $G(\mathbf{z})$, generated by the generator G from random noise $\mathbf{z}$, is real. $p_{\mathrm{data}}(\mathbf{x})$ is the distribution of real data. $p_{\mathbf{z}}(\mathbf{z})$ is the distribution of the generator's input noise.

b: WGAN LOSS

The core of the Wasserstein GAN (WGAN) [9] lies in its adoption of the Wasserstein distance, also known as Earth Mover's Distance (EMD), as the loss function for the discriminator, replacing the Jensen-Shannon divergence used in traditional GANs. The Wasserstein distance measures the "cost of transporting" one probability distribution to another, quantifying the minimal amount of work required to move all the mass of one distribution to match another. This metric provides meaningful gradient information even when the probability distributions are close or not fully overlapping.

The WGAN theorem establishes that by constructing a discriminator that satisfies the K-Lipschitz condition and maximizing its estimate of the Wasserstein distance between real and generated data, one can ensure that the training of the generator converges to a global optimum. This fundamentally addresses the issues of gradient vanishing and mode collapse prevalent in traditional GANs, resulting in a more stable training process and the generation of higher-quality samples.

WGAN-GP is an improved version of WGAN that introduces a gradient penalty term to further enhance stability. It not only uses the Wasserstein distance as the loss function for the discriminator and requires the discriminator to satisfy the K-Lipschitz continuity condition, but also incorporates a gradient penalty term. This term penalizes the expectation of the gradient norm of the discriminator's output with respect to both real and generated samples. When the gradient norm exceeds a preset threshold, an additional loss is added,

forcing the gradient of the discriminator's output to remain smooth, thereby avoiding gradient explosion or vanishing during training. is the distribution of real data. $p_{\mathbf{z}}(\mathbf{z})$ is the distribution of the generator's input noise.

c: LSGAN LOSS

LSGAN [11] builds upon the foundational principles of GANs but introduces modifications to the objective function by employing least squares loss rather than binary cross-entropy. In LSGAN, distinct loss functions are used for the generator and the discriminator. Instead of utilizing binary cross-entropy, the discriminator's loss function is based on least squares loss, which penalizes the squared difference between the discriminator's predictions and the desired targets (1 for real data and 0 for generated data). Similarly, the generator's loss function, derived from least squares loss, aims to minimize the squared difference between the discriminator's predictions for the generated samples and the target value of 1. is the distribution of real data. $p_{\mathbf{z}}(\mathbf{z})$ is the distribution of the generator's input noise.

d: CGAN LOSS

GANs faced challenges in generating images with specific attributes. To address this issue, Mehdi Mirza et al. proposed Conditional Generative Adversarial Networks (CGANs) [164], a significant advancement in the field. The core concept of CGANs involves incorporating conditional attribute information y into both the generator and the discriminator. This attribute y can encompass various forms of label information, such as the class of an image or specific facial expressions in facial images.

By integrating this attribute information, CGANs effectively convert the problem from an unsupervised learning task into a supervised learning one. This approach allows the network to better manage the learning process under controlled conditions, thereby enabling the generation of images that exhibit the desired attributes with greater precision and control. The adoption of CGANs represents a notable enhancement in the capabilities of GANs, offering a method for more accurately generating images with specific characteristics.

This development underscores the evolution of GAN architectures towards more flexible and targeted applications in image generation tasks.

e: PATCH GAN LOSS

In the CycleGAN [165] architecture, the discriminator employs a design known as "PatchGAN." Unlike the discriminator in traditional GANs, which outputs a single scalar value representing the overall quality of the entire generated image, the Patch GAN discriminator introduces a different approach. The design of Patch GAN is fully convolutional, which is why it is also referred to as a "fully convolutional GAN." In Patch GAN, the image is processed through a series of convolutional layers without being passed through fully connected layers or activation functions. Instead, the convolutional layers produce a matrix of output values that serves as a local evaluation of the image.

Each entry in this matrix, which can be True or False, corresponds to a small patch of the original image, providing localized feedback on different regions of the generated image. This approach contrasts with the traditional GAN discriminator, which uses a single scalar value to assess the entire image. By evaluating overlapping patches across the image, Patch GAN offers a more detailed and localized assessment, allowing the discriminator to focus on finer details and structural coherence. This method enhances the ability of the discriminator to capture and reflect the quality of the generated image over various spatial scales, thereby improving the overall effectiveness of the adversarial training process.

In summary, Patch GAN replaces the single global evaluation of the image with a grid of local evaluations, which better addresses the quality and consistency of the generated images. The Patch GAN design leverages the concept of a "receptive field" to analyze local regions of the image, which is a significant advantage over traditional GAN discriminators. This shift allows the discriminator to be more sensitive to local features and details, thus contributing to more robust and effective training of the generator.

f: LOSS FOR WEAKLY SUPERVISED DETECTION NETWORK

[166] To fully unleash the creative potential of the GAN network, a weakly supervised approach is employed instead of conventional supervised learning. A detection network is designed to replace and enhance the functionality of the discriminator. This detector evaluates the quality of generated images in two ways: first, by detecting any residual mask artifacts in the generated images using a deep learning network; second, by comparing the differences between the generated images and the original real images to determine if they resemble a distinguishable mask.

2) EVALUATION METRICS

a: TRANSFORMER FOR IMAGE QUALITY (TRIQ)

Transformer for Image Quality (TRIQ) [167] is the first model to use a Transformer architecture for image quality assessment. Introduced in 2020, the main idea is to first use Convolutional Neural Networks (CNNs) to extract features, followed by a shallow Transformer encoder. To handle images of different resolutions, this architecture employs adaptive positional embeddings. Considering that compressing image resolution might negatively impact image quality verification, the TRIQ framework retains the original size of the images. Initially, ResNet [7] is used as the feature extractor, with the C5 output processed through a 1×1 convolution to obtain features of dimension $H/32 \times W/32 \times 32$. Given that high-resolution images require substantial memory, a pooling layer is added before the Transformer to dynamically determine a pooling factor P. The MLP network

head after the Transformer encoder consists of two fully connected (FC) layers with a dropout layer in between, used to predict the perceptual image quality. The final output is a five-dimensional vector representing the quality distribution of the image.

b: IMAGE QUALITY TRANSFORMER (IQT)

Image Quality Transformer (IQT) [168] was proposed in 2021 and draws on the methods used in TRIQ. IQT is a Transformer-based image quality assessment method, with output results that align more closely with human perception, similar to LPIPS, for full-reference image quality assessment.

As shown in the figure, the authors use Inception-Resnet V2 [30] to extract perceptual feature representations for both the generated and reference images. The perceptual feature representation results come from the outputs of 6 intermediate layers, which are then concatenated. The feature vectors of the reference image ($\mathbf{f}_{\text{ref}}$) and the difference between the feature vectors of the reference and generated images ($\mathbf{f}_{\text{diff}}$) are input into the Transformer. Finally, the Transformer's output passes through an MLP Head to predict a final image quality score.

c: MULTI-SCALE IMAGE QUALITY TRANSFORMER(MUSIQ)

Ke et al. [169] designed a Multi-Scale Image Quality Transformer (MUSIQ) to handle native resolution images with varying sizes and aspect ratios. MUSIQ can capture image quality at different granularities through multi-scale image representations. Additionally, a novel hash-based two-dimensional spatial embedding and scale embedding were introduced to support positional embedding within the multi-scale representation.

D. SUMMARY OF LOSS FUNCTION AND EVALUATION MATRICS

There are notable differences between pixel-based, feature-based, and model-based loss functions and evaluation metrics. Pixel-based and feature-based algorithms can both be derived from first or second norm mathematical formulas or their variants. Pixel-based algorithms focus on the restored final image, while feature-based algorithms focus on the image's features. Based on this distinction, we observe that except for TV Loss, which does not need to consider the target image, all pixel-based algorithms rely on comparing the complete restored image with the complete target image. In contrast, feature-based loss functions ignore pixel-level differences, using only partial information and features to compare the restored and target images. Thus, these can be categorized into full-reference, reduced-reference, and no-reference methods. Model-based algorithms depend on deep learning models to compare the restored and target images. Due to the black-box nature of neural networks, it is challenging to determine whether this comparison is pixel-based or feature-based. Lastly, model-based losses, mainly referring to various GAN losses, require adversarial training during the generator training process. Model-based evaluation metrics, similar to feature-based ones, can directly use pre-trained models to evaluate the inpainting results.

IV. APPLICATIONS AND DATASETS

Previous research on datasets has generally been limited to general domains. However, many fields now use deep learning-based image inpainting models to accomplish tasks that would otherwise require costly human effort. To facilitate these tasks, an increasing number of datasets have been developed across various research domains. However, many of these datasets have not been adequately covered in existing reviews. In this paper, we integrate datasets with application domains, categorizing the applications of deep learning-based image inpainting into mask datasets and image datasets. Image datasets are further divided into three major categories: natural image-based, detection image-based, and artistic image-based. Natural image-based datasets refer to image data that align with the scenes visible to the human eye in everyday and ordinary real-world scenarios. These datasets can be captured using commonly available devices, such as smartphones and cameras, and include images such as portraits, landscapes, road conditions, and plant images. Detection image-based datasets refer to images captured and detected using specialized instruments for scientific research and production, such as medical images, satellite images, and infrared detection images. Artistic image-based datasets comprise images created through artistic processing and human creativity. Within each category, there are representative research areas, as illustrated in Figure 11. The classification of deep learning datasets and applications. The following sections will provide a detailed introduction to these four types of datasets.

A. MASK IMAGES

There are primarily two types of masks used in image inpainting: regular masks and irregular masks. Regular masks, mostly square or rectangular holes, are often unsuitable for practical use because real-world image damage tends to be irregular. This leads to the use of the second type, irregular masks. The two commonly used irregular mask datasets are the NVIDIA Irregular Mask dataset [20] and the Quick Draw Irregular Mask Dataset [170].

1) NVIDIA IRREGULAR MASK DATASET

Proposed by [20], contains 55,116 training masks and 12,000 test masks, with the test masks being 512 × 512 pixels in size. These masks are widely used and are categorized based on the hole size into six ranges: 1%-10%, 10%-20%, 20%-30%, 30%-40%, 40%-50%, and 50%-60%, with 2,000 masks in each category.

2) QUICK DRAW IRREGULAR MASK DATASET

Based on the Quick Draw dataset [170], consists of 10,000 training and 10,000 test binary masks, each 256 × 256 pixels in size.

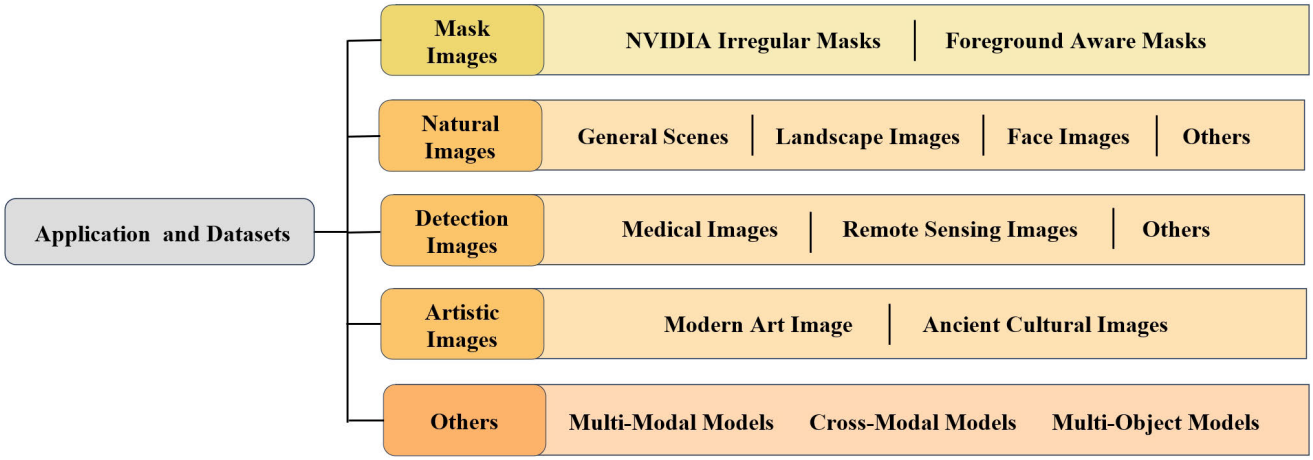

FIGURE 11. Datasets and applications of image inpainting.

B. NATURALIMAGES

Natural images refer to those captured by general photographic equipment, such as cameras or smartphones, which resemble what the human eye perceives. Datasets based on such images are the primary datasets used in image inpainting algorithms. Below, we introduce some representative datasets and applications involving natural images.

1) GENERAL SCENES

General Scenes datasets, including ImageNet [7], DIV2K [171], and LAION-5B [172], are vast and diverse, making them widely used in tasks such as image inpainting, recognition, segmentation, and multimodal applications. Research based on these datasets is extensive and will not be elaborated here.

a: IMAGENET

This dataset [7] comprises 14,197,122 images. In the early stages, most images were 224 × 224 pixels, while later additions have variable dimensions.

b: DIV2K

This dataset [171] contains 10,000 high-resolution images, each with a resolution of 2000 × 2000 pixels. It includes 800 training images, 10 validation images, and 10 test images.

c: LAION-5B

This dataset [172] contains 585,000,000 images, which is the largest publicly available text-image dataset. It is commonly used for training large multimodal models for text-guided image generation.

2) LANDSCAPE IMAGES

Landscape Images datasets, such as Paris StreetView [173], Cityscapes [174], MS COCO [175], and Places2 [176], are frequently used in image inpainting and segmentation tasks. Research based on these datasets is also extensive and will not be detailed here.

a: PARIS STREETVIEW

This dataset [173] comprises 15,000 urban scene images from Paris, with 14,900 images for training and 100 images for testing. The images have a resolution of 936 × 537 pixels.

b: CITYSCAPES

This dataset [174] includes 25,000 images, of which 5,000 have fine annotations and the remaining 20,000 have coarse annotations. The image resolution is 2048 × 1024 pixels.

c: MS COCO

[175] Primarily used for image recognition and segmentation research, this dataset consists of 328,000 images. It includes 1.5 million object instances, categorized into 80 object categories and 91 stuff categories.

d: PLACES2

This dataset [176] contains over 10 million images spanning more than 400 unique scene categories. Each category includes between 5,000 and 30,000 training images, reflecting the frequency of scenes in the real world. The images have a resolution of 256 × 256 pixels.

3) FACE IMAGES

Face Images datasets include CelebA [177], CelebA-HQ [178], and Helen Face [179], [180],. Although there is significant research on these datasets, the distinctive features of faces result in some unique applications compared to general features.

a: CELEBA

This dataset [177], released by the Multimedia Laboratory at The Chinese University of Hong Kong, is a large-scale facial

attributes dataset comprising over 200,000 celebrity images. Each image is annotated with 40 attributes.

b: CELEBA-HQ

This dataset [178] contains 35,666 high-resolution video clips (at least 512 × 512 pixels) featuring 15,653 distinct identities. Each clip is manually annotated with 83 different facial attributes, covering various aspects such as appearance, expressions, and emotions.

c: HELEN FACE

This dataset [179], [180] includes 2,330 facial images with a resolution of 400 × 400 pixels. It consists of 2,000 training images and 330 test images, all annotated with highly accurate, detailed, and consistent labels for major facial components.

Fang et al. [33] proposed a U-Net-based approach that integrates Hybrid Dilated Convolution (HDC) and Spectral Normalization to fill missing regions of any shape with sharp structures and fine detailed textures. Liu et al. [181] proposed a model that combines attention mechanisms with multi-level feature processing to maintain soft correlations with surrounding content. Li et al. [182] proposed a two-stage framework named FaceInpainter, which explicitly separates the foreground and background information of the target face to achieve controllable Identity-Guided Face Inpainting (IGFI) in heterogeneous domains. To leverage preserved identity information for face inpainting and enhance the flexibility of the model's inpainting results, Li et al. [183] proposed an Identity-specific Face Inpainting framework. This framework influences both complete and partial face inpainting under the guidance of a reference face image. It consists of an identity encoding module, a content inference module, and a generation module. The identity encoding module extracts identity embeddings from the reference image, the content inference module learns to predict the content image, and the generation module integrates the content image and reference identity embeddings to produce identity-specific inpainting results. DE-GAN [184] introduces HVAE into the generator, embedding three types of facial domain information into latent variables to guide face inpainting. This approach is the first to evaluate the problem of profile face inpainting. Wang et al. [58] proposed a novel two-stage blind face inpainting method called Frequency-Guided Transformation and Top-Down Refinement Network (FT-TDR) to address the detection of damaged regions and high-quality face inpainting. Due to the impact of the COVID-19 pandemic, Jiang et al. [185] proposed a novel mask removal and inpainting network based on pre-known facial attributes (including nose, chubby, makeup, gender, mouth, beard, and young). This network employs a GAN-based dual-stream architecture to ensure that the restored facial images closely resemble real appearances. Yu et al. [186] designed a two-stage network, employing a coarse-to-fine approach, using another reference image with the same identity as the masked input as a conditional input to achieve detailed face inpainting. To reconstruct a face based on the periocular region (eyes to face), Hassanpour et al. [187] proposed a GAN-based model called Eyes To Face GAN (E2F-GAN). This model comprises two main modules: a coarse module and a refinement module, producing realistic and semantically coherent images. Deep learning networks have been widely demonstrated to capture the semantics of facial expressions effectively. However, during high-resolution face image inpainting, networks often face convergence issues. To address this, He et al. [188] designed a high-definition face inpainting model based on pre-trained ResNet [7] and StyleGAN. To improve inpainting performance and reduce computational complexity during the inference process, Xu et al. [189] proposed combining Parallel Visual Attention (PVA) with a diffusion model. They demonstrated that PVA not only provides effective language controllability but also ensures strong identity preservation. Furthermore, compared to custom diffusion methods, PVA requires only 40 fine-tuning steps per new identity, resulting in a significant speed improvement of over 20 times. To address the issue of face recognition difficulties caused by frequent mask-wearing during the COVID-19 pandemic, Ma et al. [190] proposed a self-supervised Siamese inference network. This approach enhances the generalization capability and robustness of the encoder by reconstructing the facial features covered by masks, significantly improving masked face recognition performance. To protect the privacy of portraits in online photos, Sun et al. [116] proposed a novel head inpainting and obfuscation technique. This method first generates facial landmarks from the image context (such as body posture) to seamlessly infer perceptible head poses, followed by head inpainting based on the facial landmark conditions. Similarly aiming to achieve privacy protection, Upenik et al. [191] proposed an image inpainting network that reversibly removes user-selected content. This method improves privacy protection by allowing users to hide objects within the overall image with minimal distortion.

4) OTHERS

In addition to the mainstream datasets mentioned above, there are other natural image datasets and applications, such as:

a: SVNH

This dataset [192] includes 99,289 images of street view house numbers, with 73,257 digits for training and 26,032 digits for testing. Additionally, there are 531,131 extra samples of slightly lower difficulty, used as additional training data.

b: STANFORD CARS

[193] Primarily used for fine-grained classification tasks, this dataset contains 16,185 images of cars of different models, with 8,144 images for training and 8,041 images for testing.

c: FAÇADE

This dataset [194] includes 606 rectified images from various sources, manually annotated. The facades come from cities around the world and represent different architectural styles.

d: DTD

The Describable Textures Dataset [195] is a texture database comprising 5,640 images organized according to 47 human-perceived categories. Each category contains 120 images. Image sizes range from 300 × 300 to 640 × 640 pixels, with each image representing at least 90% of the surface attributes of the category.

e: SUNCG

This dataset [196] contains over 45,000 3D scenes, with each scene segmented into individual rooms and labeled with one or more room types, such as bedroom, living room.

Grigorev et al. [197] proposed a novel deep learning approach that learns body surface textures from partial photographs and leverages pose guidance to achieve the re-synthesis of full-body images. Cui et al. [198] proposed a flexible character generation framework known as ''Dress in Order'' (DiOr), which supports 2D pose transfer, virtual try-on, and various fashion editing tasks. The core innovation of DiOr is a novel sequential generation pipeline that sequentially dresses a character, enabling different appearances of the same garment when worn in different orders. To restore severely degraded old photographs, Wan et al. [199] proposed a triple-domain translation network that integrates two Variational Autoencoders (VAEs) and leverages both real photographs and a large volume of synthetic image pairs. Chen et al. [200] developed an image inpainting technique to remove soil or other obstructions from photos of plant roots in transparent root boxes, facilitating better segmentation of the roots from the soil background. This technique restores gaps in disconnected root segments. Due to the challenges in license plate recognition caused by camera exposure and license plate blurring, Cheng et al. [201] proposed a method using the cGAN network structure from the Pix2Pix [157] framework to recover clear license plates from dynamically blurred ones. Wan et al. [202] proposed a single-image reflection removal algorithm and provided the dataset ''SIR 2+.'' They conducted quantitative and visual quality comparisons of state-of-the-art single-image reflection removal algorithms and discussed outstanding issues for improving reflection removal techniques.

C. DETECTION IMAGES

Detection images refer to a series of images obtained through instruments that capture data beyond the capabilities of the human eye, requiring specific imaging techniques. Common examples include medical images, remote sensing images, and Scanning Acoustic Microscopy (SAM) images.

1) MEDICAL IMAGES

With the advancement of medical technology and the integration of artificial intelligence, a series of medical image datasets have been established. These datasets often come with usage restrictions or are only available for internal use within institutions. Here, I introduce some publicly available medical image datasets on the Kaggle platform:

a: DRIVE DATASET

This is a digital retinal image dataset for vessel segmentation, containing 40 photographs. Among these, 7 images show early signs of mild diabetic retinopathy.

b: SCR DATASET

This dataset contains segmented chest X-rays, intended for comparative studies of the segmentation of lung fields, heart, and clavicles in standard posteroanterior chest radiographs.

c: CARDIAC MRI DATASET

This dataset comprises atrial medical images of heart disease patients, including 33 patient cases. Each subject's sequence consists of 20 frames and 8-15 slices, totaling 7,980 images.

d: NIH DATASET

The Chest X-ray dataset from the National Institutes of Health contains 30,805 patient images with 14 disease labels. Each image can have multiple labels, which are automatically extracted from the associated radiology reports.

Various factors can lead to the degradation or unavailability of medical images. To address this issue, several methods have been proposed: Armanious et al. [203] introduced a method for medical image inpainting by combining two patch-based Generative Adversarial Networks (GANs). Magnetic Resonance Imaging (MRI) images, acquired as multi-layer two-dimensional (2D) images, often suffer from artifacts when reformatted in orthogonal planes. To restore these damaged MRI regions, Chai et al. [204] treated the corrupted image data or missing slices as an image mask and proposed an edge-guided Generative Adversarial Network to recover brain MRI images. For spinal Computed Tomography (CT) image inpainting, Miao et al. [205] first segmented spinal CT images using a neural network to obtain binary images. These binary images were then restored using a GAN. To overcome the degradation problem of deep networks, Wang et al. [206] proposed a medical image inpainting model based on edge and structural information (ESMII), which has been proven effective in restoring COVID-19 CT, abdominal CT, and abdominal MRI data. To address issues such as vascular inpainting in medical images, removal of specular reflections in endoscopic images, and removal of MRI artifacts, Gapon et al. [207] employed a method that combines searching various sizes of patches and pre-trained neural networks. Xie et al. [208] addressed the problem of truncated regions in CT images by using a Generative Adversarial Network with gated

convolution (GatedConv). These inpainted images were then applied for dose calculation in radiotherapy.

2) REMOTE SENSING IMAGES

a: NWPU VHR-10 DATASET

This dataset is used for spatial object detection in very high resolution remote sensing images. It contains 800 images, including 650 target-containing images and 150 background images. The dataset covers 10 target categories: airplanes, ships, oil tanks, baseball fields, tennis courts, basketball courts, athletic fields, ports, bridges, and cars.

b: RSOD DATASET

This open dataset is designed for object detection in remote sensing images. It includes 446 images with 4,993 airplanes, 165 images with 1,586 oil tanks, 189 images with 191 sports fields, and 176 images with 180 overpasses.

c: DIOR DATASET

The DIOR (Dataset for Object Detection in Optical Remote Sensing Images) is a large-scale benchmark dataset for object detection in optical remote sensing images. It comprises 23,463 images and 192,472 instances, covering 20 object classes.

d: DOTA DATASET

The DOTA (Dataset for Object Detection in Aerial Images) is a large-scale dataset for object detection in aerial images. It includes 2,806 aerial images captured from various sensors and platforms, designed for developing and evaluating object detection algorithms.

e: HRRSD DATASET

Released by the University of Chinese Academy of Sciences in 2019, the HRRSD (High-Resolution Remote Sensing Dataset) contains 21,761 images sourced from Google Earth and Baidu Maps. The spatial resolution ranges from 0.15 meters to 1.2 meters. The dataset includes 55,740 instances across 13 categories, with approximately 4,000 instances per category.

Remote sensing images have significant applications in scene classification, semantic Segmentation, and object detection [209], but missing data reconstruction is a classic yet challenging problem in remote sensing imagery. To address the limitations of traditional convolutional neural network-based methods, which often require supplementary data and can only handle specific tasks, Shao et al. [210] proposed a hybrid network combining convolutional and attention mechanisms based on Generative Adversarial Networks (GANs). This approach enables the model to generate coherent structures with better detail. Very High Resolution (VHR) satellite and aerial images are often affected by scene occlusion caused by redundant objects. To address this, a reconstruction network divided into two parts–structure prediction and texture generation–has been proposed to remove redundant objects. Very High Resolution (VHR) satellite and aerial images are often affected by scene occlusion caused by redundant objects. To address this issue, Xu et al. [211] proposed a reconstruction network divided into two parts: structure prediction and texture generation, which effectively removes redundant objects.

Cloud contamination significantly limits the potential use of optical remote sensing images in Earth science applications. Numerous solutions have been developed to remove clouds from multispectral images. Among these: Li et al. [212] proposed a semi-supervised method based on Generative Adversarial Networks (GAN) and a cloud distortion physical model (CR-GAN-PM) for removing thin clouds from unpaired images in different regions. Zi et al. [213] introduced a method for thin cloud removal from multispectral images, combining traditional approaches with deep learning using the U-Net method. Heng Pan [214] proposed a model named Spatial Attention Generative Adversarial Network (SpA-GAN), which utilizes local-to-global spatial attention to identify and focus on cloud regions, generating higher-quality cloud-free images. Darbaghshahi et al. [215] employed two Generative Adversarial Networks (GANs) to convert SAR images to optical images and then remove clouds from the converted images.

3) OTHERS

Somani et al. [46] employed hypergraph image inpainting techniques to fill in missing information, aiming to improve the resolution of SAM images, demonstrating that combining SAM with hypergraphs can produce noise-robust interpretations. In SAM, the frequency of excitation signals, signal-to-noise ratio (SNR), and pixel size all play crucial roles in determining acoustic image resolution. Pragyan Banerjee et al. [216] have proposed a deep learning-based image inpainting technique for acoustic microscopy. This method employs various Generative Adversarial Networks (GANs) to repair gaps in the original images and generate fourfold enhanced images.Electrical imaging logging data is crucial for identifying lithology and sedimentary facies in complex reservoirs. However, due to various factors in practical logging operations, electrical imaging logging often fails to achieve full borehole coverage. To address this, Wu et al. [217] applied the EdgeConnect model [16] to restore electrical imaging logs, resulting in the recovery of complex reservoir images with unique textures and high heterogeneity.

D. ARTISTIC IMAGES

Natural images and detection images are refined representations of the objective world, created without the influence of emotions or humanistic thoughts. In contrast, artistic images are always imbued with unique emotions and full passion. Through the long-established principles of artistic creation and stylized processing, paintings possess a humanistic

richness and artistic quality that other types of images cannot match.

1) WIKIART

WikiArt has archived approximately 250,000 artworks by around 3,000 artists, and has been localized into 8 languages. These artworks are housed in museums, universities, city halls, and other municipal buildings across more than 100 countries.

2) ART IMAGES-9K DATASET

This dataset comprises approximately 9,000 art images spanning 5 types of art. The images were sourced from Google Images, Yandex Images, and the Russian Museum. The dataset is divided into training and testing sets and includes the following art types: watercolor paintings, oil paintings, sculptures, graphic arts, and iconography (ancient Russian art).

3) MODERN ART IMAGES

Xie et al. [99] proposed a network for repairing comics, which divides the complex inpainting process into two main stages: semantic repair and appearance synthesis. In the semantic repair network, two decoders operate in parallel. In the second stage, the appearance synthesis involves two encoders running in parallel.

4) ANCIENT CULTURAL IMAGES

Tianxiu Yu's et al. [218] utilized a inpainting model based on Partial Convolutions U-Net to repair 10,000 slices of Dunhuang murals, each sized at 256 × 256 pixels. In this study, dust-like and jelly-like masks were specifically designed to better simulate mural defects in realistic scenarios. Jianfang Cao's et al. [219] incorporated an FCN structure into the GLCIC model, using a dataset of 12,000 non-fixed-sized Wutaishan murals for the inpainting study. Irina-Milaela Ciortan et al. [220] used over 5,600 original Dunhuang mural images sized at 256 × 256 pixels. After data augmentation, they applied a two-stage progressive inpainting network to repair edges initially and then proceeded with color filling and inpainting. This study, building upon Tianxiu Yu's research, introduced the use of skeletonization methods to simulate irregular linear defects such as peeling and cracking in mural masks. Nianyi Wang et al. [221] worked with 2,782 Thangka mural images sized at 512 × 512 pixels. After data augmentation, this dataset was used to design a Partial Convolutions U-Net network with multi-scale convolution kernels for the inpainting of random brushstroke line defects and random small-area block defects.

E. MULTIMODAL, CROSS-MODAL, AND MULTI-OBJECT IMAGE INPAINTING MODELS

In addition to traditional image inpainting tasks, new applications based on multimodal, cross-modal, and multi-object image inpainting have gradually emerged.

[222] demonstrated that treating the problem as a simple image inpainting task can be surprisingly effective if the inpainting algorithm is trained on the correct data. They trained a masked autoencoder on 88,000 unlabeled images sourced from Arxiv academic papers and applied visual prompts to these pre-trained models. This approach yielded strong results across various downstream image tasks, including foreground segmentation, single-object detection, colorization, and edge detection. The Inpainting Anything [223] project, built upon the Segment Anything Model (SAM) [224], begins by using PP-YOLOE [225] to detect all objects in a video. The bounding boxes of the objects that the user wishes to remove are then sequentially input into the SAM model to obtain the mask for each object. All object masks are combined into a final mask, which is fed into the LaMa [226] model for object removal. This project implements three image editing modes within Inpaint Anything: Remove Anything, Fill Anything, and Replace Anything. Additionally, it extends the Remove Anything functionality to videos through the Remove Anything Video mode. PowerPaint [227] introduces learnable task prompts and targeted fine-tuning strategies to guide the model in focusing on different inpainting objectives. It also employs negative prompts to effectively remove unwanted objects. Additionally, prompt interpolation techniques are used to achieve controllable shape-guided inpainting. [228] developed a system called Visual ChatGPT, which integrates various Visual Foundation models, allowing users to interact with ChatGPT in the following ways: 1) by sending and receiving not only text but also images; 2) by providing complex visual queries or editing instructions that require multi-step collaboration among multiple AI models; 3) by giving feedback and requesting corrections to the results. Considering models with multiple inputs/outputs and those requiring visual feedback, the authors designed a series of prompts to inject visual model information into ChatGPT. Experiments show that Visual ChatGPT opens new avenues for exploring the visual capabilities of ChatGPT with the aid of Visual Foundation models. [229] proposed an image inpainting model that integrates YOLOv8 [230] with LaMa [226]. YOLOv8 is used to identify the areas that require inpainting, enhancing both accuracy and speed. LaMa then generates content that is contextually relevant to the surrounding regions. The iEdit [231] model introduces a novel text-guided image editing approach based on source images and text prompts. This method is developed on the LAION 5B dataset and combines weak supervision with contrastive learning.

F. SUMMARY OF DATA AND APPLICATIONS

In image inpainting, the data used primarily consists of masked data and complete image data. These two elements are sufficient to generate paired data for model training and learning. Compared to image restoration tasks such as deblurring, denoising, dehazing and super-resolution [232],

it is easier to simulate defective data in masked datasets and pair it with the original ground truth images for training. This is because models in other domains must autonomously identify and handle various types of disturbances and noise distributions to meet real-world scenarios. In contrast, in image inpainting research, it is often possible to directly inform the model of the specifics of the mask, and even if the mask does not precisely match the missing or to-be-removed regions, inpainting can still be achieved. Of course, this does not exclude the high cost of manually annotating masks in certain special cases. Nonetheless, generating paired data for image inpainting is significantly easier than simulating complex scenarios involving different types of noise, shadows, and other artifacts. This explains why there is more research on unsupervised, imbalanced data, weakly supervised, and zero-shot learning in the fields of image restoration, such as denoising and super-resolution. Moreover, although masked data for image inpainting is relatively easy to simulate, inpainting tasks often face severe information and feature loss. This leads to challenges in accurately matching latent space vectors in unsupervised or zero-shot learning models, particularly when the missing area is large. Consequently, such models rely more heavily on the quality of pre-trained models to achieve satisfactory results, which is why research on unsupervised and zero-shot image inpainting models is relatively scarce.

In this study, the complete image datasets used for image inpainting are categorized into three types: natural image-based, detection image-based, and artistic image-based. Among these, natural image-based datasets are currently the most abundant and readily available. As a result, most research on image inpainting models is based on this type of data. Correspondingly, only these large-scale datasets can support the training of complex, large models without leading to overfitting. Consequently, most studies on multimodal, cross-modal, and large-model image inpainting are trained on natural image-based datasets. Detection image-based datasets play a significant role in technology and production; however, unlike natural image-based datasets, they require specialized equipment and specific scenarios for collection. Thus, these datasets are relatively scarce, and the characteristics differ significantly across different fields, limiting model training. Therefore, many studies have employed small models or large pre-trained models to perform inpainting on such damaged images. Moreover, in practical applications, detection image-based datasets often have strict requirements and standards for inpainting, demanding a single accurate inpainting result rather than multiple potential outcomes. Artistic image-based datasets include both modern artistic images and ancient artifact images. The latter, due to difficulties in preservation and their irreplaceable nature, are scarce, though the demand for inpainting is high and the inpainting results must strictly adhere to historical accuracy. Consequently, most researchers have opted for single-result inpainting using small models for inpainting. In contrast, modern artistic image inpainting often involves image editing, which accommodates multiple inpainting outcomes. Overall, artistic image-based datasets differ from the other two types in that their features do not originate from the real world; they may exhibit exaggerated colors and abstract textures, presenting substantial differences compared to the other datasets. Therefore, many studies in this area have drawn upon research in style transfer.

V. CHALLENGES AND FUTURE DIRECTION

Based on the extensive work of many researchers, deep learning-based models have achieved significant progress in image inpainting. However, several challenges and limitations remain, providing directions for future research.

1) SEMANTIC UNDERSTANDING OF IMAGES

a: CHALLENGE

Current models still lack a deep understanding of high-level semantic information within images. Although recent image generation models have achieved impressive results, there remains a disparity between these models and inpainting models. Inpainting models are constrained by the available image information, which requires not only continuity and smoothness in lines and textures but also logical coherence in the high-level semantics of the image.

b: FUTURE

In light of these challenges, the future development of inpainting models will undoubtedly be accompanied by an enhanced understanding of image semantics. This remains a core challenge in the field of computer vision, which involves extracting high-level semantic information from images. To bridge the gap between a computer's understanding of images and human perception, several strategies can be employed. Firstly, improving the quality and quantity of data is crucial. Secondly, integrating prior knowledge, such as common sense, physical laws, and semantic knowledge graphs, into image semantic understanding models can help the models better comprehend complex scenes and handle semantic ambiguities. This, in turn, will enable inpainting models to more accurately infer and predict hidden information within scenes.

2) INTEGRATION OF MULTIPLE DATA SOURCES

a: CHALLENGE

There is a lack of training that integrates multiple data sources. Existing image inpainting models, when applied to various specialized fields, are typically retrained on specific datasets from those fields. However, due to the significant differences in characteristics between image datasets, models trained on natural images struggle to generate reasonable styles when tested on art image datasets, and models trained on medical images find it difficult to repair natural images. This results in a lack of generalization across fields, necessitating individual training for each domain, which is costly. Furthermore, image data in niche fields are often

scarce, making it challenging to achieve good inpainting results.

b: FUTURE

In response to these challenges, one future direction is the development of hybrid models that possess both generality and specificity. Structurally, this can be achieved by balancing inter-domain trade-offs through shared parameters and branched networks. Additionally, training a general model by incorporating diverse data from different domains can lead to superior performance across multiple domains. Moreover, domain adaptation and transfer learning can be utilized by pretraining models on one domain or a large-scale dataset, followed by fine-tuning for specific domains, thus enhancing the model's cross-domain generalization capabilities.

3) MULTI-OBJECTIVE MODELS

a: CHALLENGE

There is a need for models that can simultaneously perform defect detection, inpainting, denoising, and enhancement. Most current image inpainting models simulate defects, but these simulated defects do not comprehensively cover the variety of actual image defects encountered in practice. Therefore, manual annotation of defects in each image is often required, which is labor-intensive and costly. If models could combine defect detection and inpainting, they would be more suitable for practical applications.

b: FUTURE

Given these challenges, future research directions will likely focus on end-to-end multi-objective image restoration that integrates defect detection, inpainting, denoising, and enhancement. One approach could involve processing multiple tasks within the same model, improving performance on each task by sharing network structures or weights. On one hand, feature extractors could be shared, capturing the underlying features of images with a shared extractor, followed by different task-specific branches handling the various objectives. On the other hand, the intrinsic relationships between tasks could be leveraged by designing loss functions that optimize all tasks simultaneously or balancing the contributions of different tasks through task weighting. Another approach could involve model ensembling, where models targeting different tasks are integrated via techniques like weighting or stacking, thereby enhancing overall performance.

4) MULTI-MODAL, CROSS-MODAL BASED ON LARGE MODELS

a: CHALLENGE

The fusion of multi-modal data, such as images, text, and audio, is highly complex. Different modalities possess distinct feature spaces and distributions; for instance, there is significant semantic disparity between images and text. Effectively integrating these heterogeneous data within the same model poses a challenge. In multi-modal and cross-modal inpainting tasks, the generated content must maintain semantic consistency. Furthermore, large multi-modal and cross-modal models are typically very large, and the training and inference processes demand significant computational resources, which presents challenges for computational efficiency and resource management.

b: FUTURE

To address these challenges, further development and optimization of multi-modal joint learning models are necessary, enabling inpainting tasks to fully utilize information from various modalities. This can be achieved by fine-tuning on pre-trained, well-performing large-scale general multi-modal and cross-modal models and then adapting them to understand image inpainting tasks in conjunction with other types of information, thereby achieving higher performance and generalization.

5) DATA PRIVACY AND SECURITY

a: CHALLENGE

With the increasing focus on data privacy, future image inpainting technology needs to ensure privacy security. This includes developing privacy-preserving image inpainting techniques, such as federated learning and differential privacy, to ensure model training and inpainting can be completed without data leaving the local environment. Additionally, secure data processing must be ensured to protect sensitive data in fields such as healthcare and surveillance, preventing data breaches and misuse.

b: FUTURE

In light of these challenges, incorporating more explainability and transparency mechanisms into inpainting models will be crucial, making the inpainting process and results more transparent, especially when applied in sensitive areas such as medical image inpainting and legally relevant image processing. As image inpainting and generation technologies continue to advance, corresponding advancements in image generation detection technologies are expected to follow suit.

In conclusion, while deep learning-based image inpainting has achieved substantial advancements, addressing these limitations will pave the way for more robust, generalizable, and secure models, driving further progress in the field.

VI. CONCLUSION

This paper systematically reviews deep learning-based image inpainting algorithms, providing an innovative organization and classification of these models based on their generator strategies and the loss functions employed. In terms of generator strategies, we further subdivide them based on model components into layer-based, connection-based, multi-network combinations, and multi-strategy inpainting approaches, thoroughly analyzing the application and effectiveness of these strategies in image inpainting.

Simultaneously, concerning loss function strategies, we delve into a detailed discussion by combining and comparing these with evaluation metrics. We categorize these methods into pixel-based, feature-based, and model-based approaches, providing a comprehensive analysis of their impact on image inpainting performance, thus offering robust tools for algorithm performance evaluation.

Additionally, we classify image inpainting applications into three main types–natural image inpainting, detection image inpainting, and artistic image inpainting based on the characteristics of the datasets. We further analyze the applications of inpainting in these respective fields, offering a deeper understanding of the unique challenges and characteristics associated with image inpainting across different domains.

In summary, this paper not only comprehensively covers existing deep learning-based image inpainting algorithms but also addresses the issues present in previous review studies, such as unclear classification, incomplete algorithm coverage, and a lack of systematic analysis of loss functions and evaluation metrics. This work provides strong support and guidance for further research and development in the field of image inpainting.

REFERENCES

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. 27th Annu. Conf. Comput. Graph. Interact. Techn.*, 2000, pp. 417–424.

[2] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212, Sep. 2004.

[3] J. Shen and T. F. Chan, "Mathematical models for local nontexture inpaintings," *SIAM J. Appl. Math.*, vol. 62, no. 3, pp. 1019–1043, Jan. 2002.

[4] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, "PatchMatch: A randomized correspondence algorithm for structural image editing," *ACM Trans. Graph.*, vol. 28, no. 3, p. 24, May 2009.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9.

[6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[7] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.

[9] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 214–223.

[10] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of Wasserstein GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[11] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2813–2821.

[12] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2536–2544.

[13] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–14, Aug. 2017.

[14] Y. Wang, X. Tao, X. Qi, X. Shen, and J. Jia, "Image inpainting via generative multi-column convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–10.

[15] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5505–5514.

[16] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi, "EdgeConnect: Structure guided image inpainting using edge prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 3265–3274.

[17] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.

[18] Z. Yan, X. Li, M. Li, W. Zuo, and S. Shan, "Shift-Net: Image inpainting via deep feature rearrangement," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 1–17.

[19] X. Hong, P. Xiong, R. Ji, and H. Fan, "Deep fusion network for image completion," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 2033–2042.

[20] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 85–100.

[21] Y. Zeng, J. Fu, H. Chao, and B. Guo, "Learning pyramid-context encoder network for high-quality image inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1486–1494.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[23] Y. Zhou, C. Barnes, E. Shechtman, and S. Amirghodsi, "TransFill: Reference-guided image inpainting by merging multiple color and spatial transformations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 2266–2267.

[24] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 15979–15988.

[25] Q. Dong, C. Cao, and Y. Fu, "Incremental transformer structure enhanced image inpainting with masking positional encoding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11348–11358.

[26] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 6840–6851.

[27] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," 2020, *arXiv:2010.02502*.

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[29] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[30] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," in *Proc. AAAI Conf. Artif. Intell.*, 2017, vol. 31, no. 1, pp. 1–7.

[31] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*.

[32] M.-C. Sagong, Y.-G. Shin, S.-W. Kim, S. Park, and S.-J. Ko, "PEPSI: Fast image inpainting with parallel decoding network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11352–11360.

[33] Y. Fang, Y. Li, X. Tu, T. Tan, and X. Wang, "Face completion with hybrid dilated convolution," *Signal Process., Image Commun.*, vol. 80, Feb. 2020, Art. no. 115664.

[34] T. A. Mohite and G. S. Phadke, "Image inpainting with contextual attention and partial convolution," in *Proc. Int. Conf. Artif. Intell. Signal Process. (AISP)*, Jan. 2020, pp. 1–6.

[35] L. Nie, W. Yu, S. Li, Z. Zhang, N. Jiang, X. Zhang, and J. Gong, "Progressive inpainting strategy with partial convolutions generative networks (ppcgn)," in *Proc. 28th Int. Conf.*, Sanur, Indonesia. Springer, Dec. 2021, pp. 640–647.

[36] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. Huang, "Free-form image inpainting with gated convolution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4470–4479.

[37] Z. Yi, Q. Tang, S. Azizi, D. Jang, and Z. Xu, "Contextual residual aggregation for ultra high-resolution image inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7505–7514.

[38] F. Wang, W. Li, Y. Liu, Y. Gong, Z. Gao, and J. Lu, "Face inpainting combining structured forest edge information and gated convolution," in *Proc. 3rd Int. Conf. Natural Lang. Process. (ICNLP)*, Mar. 2021, pp. 213–217.

[39] C. Cao and Y. Fu, "Learning a sketch tensor space for image inpainting of man-made scenes," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14489–14498.

[40] X. Ma, Y. Deng, L. Zhang, and Z. Li, "A novel generative image inpainting model with dense gated convolutional network," *Int. J. Comput. Commun. Control*, vol. 18, no. 2, Apr. 2023.

[41] Y. Shen, Y. Su, L. Wang, and D. Jia, "Research on image inpainting algorithms based on attention guidance," *J. Adv. Comput. Intell. Intell. Informat.*, vol. 27, no. 2, pp. 190–197, Mar. 2023.

[42] Y. Ma, X. Liu, S. Bai, L. Wang, D. He, and A. Liu, "Coarse-to-fine image inpainting via region-wise convolutions and non-local correlation," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 3123–3129.

[43] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[44] S. Bai, F. Zhang, and P. H. S. Torr, "Hypergraph convolution and hypergraph attention," *Pattern Recognit.*, vol. 110, Feb. 2021, Art. no. 107637.

[45] G. Wadhwa, A. Dhall, S. Murala, and U. Tariq, "Hyperrealistic image inpainting with hypergraphs," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 3911–3920.

[46] A. Somani, P. Banerjee, K. Agarwal, M. Rastogi, D. K. Prasad, and A. Habib, "Image inpainting with hypergraphs for resolution improvement in scanning acoustic microscopy," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2023, pp. 3113–3122.

[47] H.-Y. Li, L.-C. Xiong, L. Guo, and H.-J. Li, "Two-stage inpainting algorithm based on U-Net edge generation and hypergraphs convolution," *J. Northeastern Univ.*, vol. 44, no. 3, p. 331, 2023.

[48] T. Cheng, T. Bi, W. Ji, and C. Tian, "Graph convolutional network for image restoration: A survey," *Mathematics*, vol. 12, no. 13, p. 2020, Jun. 2024.

[49] L. Jiao, J. Chen, F. Liu, S. Yang, C. You, X. Liu, L. Li, and B. Hou, "Graph representation learning meets computer vision: A survey," *IEEE Trans. Artif. Intell.*, vol. 4, no. 1, pp. 2–22, Feb. 2023.

[50] Y. Song, C. Yang, Z. Lin, X. Liu, Q. Huang, H. Li, and C.-C. J. Kuo, "Contextual-based image inpainting: Infer, match, and translate," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[51] N. Wang, J. Li, L. Zhang, and B. Du, "MUSICAL: Multi-scale image contextual attention learning for inpainting," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 3748–3754.

[52] N. Wang, S. Ma, J. Li, Y. Zhang, and L. Zhang, "Multistage attention network for image inpainting," *Pattern Recognit.*, vol. 106, Oct. 2020, Art. no. 107448.

[53] Y. Zeng, Z. Lin, J. Yang, J. Zhang, E. Shechtman, and H. Lu, "High-resolution image inpainting with iterative confidence feedback and guided upsampling," in *Proc. 16th Eur. Conf.*, Glasgow, U.K. Springer, Aug. 2020, pp. 1–17.

[54] L. Zhao, Q. Mo, S. Lin, Z. Wang, Z. Zuo, H. Chen, W. Xing, and D. Lu, "UCTGAN: Diverse image inpainting based on unsupervised cross-space translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5740–5749.

[55] C. Xie, S. Liu, C. Li, M.-M. Cheng, W. Zuo, X. Liu, S. Wen, and E. Ding, "Image inpainting with learnable bidirectional attention maps," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8857–8866.

[56] J. Qin, H. Bai, and Y. Zhao, "Multi-scale attention network for image inpainting," *Comput. Vis. Image Understand.*, vol. 204, Mar. 2021, Art. no. 103155.

[57] T. Wang, D. Xiang, C. Yang, J. Liang, and C. Shi, "NLKFill: High-resolution image inpainting with a novel large kernel attention," *Complex Intell. Syst.*, vol. 10, no. 4, pp. 4921–4938, Aug. 2024.

[58] J. Wang, S. Chen, Z. Wu, and Y.-G. Jiang, "FT-TDR: Frequency-guided transformer and top-down refinement network for blind face inpainting," *IEEE Trans. Multimedia*, vol. 25, pp. 2382–2392, 2022.

[59] W. Huang, Y. Deng, S. Hui, Y. Wu, S. Zhou, and J. Wang, "Sparse self-attention transformer for image inpainting," *Pattern Recognit.*, vol. 145, Jan. 2024, Art. no. 109897.

[60] S. Chen, A. Atapour-Abarghouei, and H. P. H. Shum, "HINT: High-quality INpainting transformer with mask-aware encoding and enhanced attention," *IEEE Trans. Multimedia*, vol. 26, pp. 7649–7660, 2024.

[61] W. Wang, J. Dai, Z. Chen, Z. Huang, Z. Li, X. Zhu, X. Hu, T. Lu, L. Lu, H. Li, X. Wang, and Y. Qiao, "InternImage: Exploring large-scale vision foundation models with deformable convolutions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 14408–14419.

[62] C. Zheng, T.-J. Cham, J. Cai, and D. Phung, "Bridging global context interactions for high-fidelity image completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11502–11512.

[63] P. Shamsolmoali, M. Zareapoor, and E. Granger, "TransInpaint: Transformer-based image inpainting with context adaptation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, vol. 30, Oct. 2023, pp. 849–858.

[64] C. Cao, Q. Dong, and Y. Fu, "ZITS++: Image inpainting by improving the incremental transformer on structural priors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 10, pp. 12667–12684, Oct. 2023.

[65] K. Ko and C.-S. Kim, "Continuously masked transformer for image inpainting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 13123–13132.

[66] M. Zhou, X. Liu, T. Yi, Z. Bai, and P. Zhang, "A superior image inpainting scheme using transformer-based self-supervised attention GAN model," *Exp. Syst. Appl.*, vol. 233, Dec. 2023, Art. no. 120906.

[67] J. Wu, Y. Feng, H. Xu, C. Zhu, and J. Zheng, "SyFormer: Structure-guided synergism transformer for large-portion image inpainting," in *Proc. AAAI Conf. Artif. Intell.*, 2024, vol. 38, no. 6, pp. 6021–6029.

[68] W. Miao, L. Wang, H. Lu, K. Huang, X. Shi, and B. Liu, "ITrans: Generative image inpainting with transformers," *Multimedia Syst.*, vol. 30, no. 1, Feb. 2024.

[69] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8162–8171.

[70] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," Tech. Rep., 1985.

[71] R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6882–6890.

[72] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. Metaxas, "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5908–5916.

[73] L. Liao, R. Hu, J. Xiao, and Z. Wang, "Edge-aware context encoder for image inpainting," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2018, pp. 3156–3160.

[74] Y. Song, C. Yang, Y. Shen, P. Wang, Q. Huang, and C.-C. J. Kuo, "SPG-net: Segmentation prediction and guidance network for image inpainting," 2018, *arXiv:1805.03356*.

[75] H. V. Vo, N. Q. K. Duong, and P. Pérez, "Structural inpainting," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 1948–1956.

[76] Y. Wang and J. Shin, "SFRNet: A deep three-stage identity and structure feature refinement network for facial image inpainting," *KSII Trans. Internet Inf. Syst.*, vol. 17, no. 3, 2023.

[77] C. Dong, H. Liu, X. Wang, and X. Bi, "Image inpainting method based on AU-GAN," *Multimedia Syst.*, vol. 30, no. 2, p. 101, Apr. 2024.

[78] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.

[79] Z. Guo, Z. Chen, T. Yu, J. Chen, and S. Liu, "Progressive image inpainting with full-resolution residual network," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 2496–2504.

[80] S. Yang, Y. Wang, H. Cai, and X. Chen, "Residual inpainting using selective free-form attention," *Neurocomputing*, vol. 510, pp. 149–158, Oct. 2022.

[81] H. Luo and Y. Zheng, "Semantic residual pyramid network for image inpainting," *Information*, vol. 13, no. 2, p. 71, Feb. 2022.

[82] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, and H. Li, "High-resolution image inpainting using multi-scale neural patch synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4076–4084.

[83] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1395–1403.

[84] J. Li, F. He, L. Zhang, B. Du, and D. Tao, "Progressive reconstruction of visual structure for image inpainting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5961–5970.

[85] Y. Luo and R. Duraiswami, "Canny edge detection on NVIDIA CUDA," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2008, pp. 1–8.

[86] W. Xiong, J. Yu, Z. Lin, J. Yang, X. Lu, C. Barnes, and J. Luo, "Foreground-aware image inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5833–5841.

[87] H. Shao, Y. Wang, Y. Fu, and Z. Yin, "Generative image inpainting via edge structure and color aware fusion," *Signal Process., Image Commun.*, vol. 87, Sep. 2020, Art. no. 115929.

[88] J. Yang, Z. Qi, and Y. Shi, "Learning to incorporate structure knowledge for image inpainting," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 12605–12612.

[89] W. Gao, X. Zhang, L. Yang, and H. Liu, "An improved Sobel edge detection," in *Proc. 3rd Int. Conf. Comput. Sci. Inf. Technol.*, vol. 5, Jul. 2010, pp. 67–71.

[90] H. Roy, S. Chaudhury, T. Yamasaki, and T. Hashimoto, "Image inpainting using frequency-domain priors," *J. Electron. Imag.*, vol. 30, no. 2, Apr. 2021, Art. no. 023016.

[91] X. Guo, H. Yang, and D. Huang, "Image inpainting via conditional texture and structure dual generation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14114–14123.

[92] Y. Yamashita, K. Shimosato, and N. Ukita, "Boundary-aware image inpainting with multiple auxiliary cues," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 618–628.

[93] Y. Ren, X. Yu, R. Zhang, T. H. Li, S. Liu, and G. Li, "StructureFlow: Image inpainting via structure-aware appearance flow," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 181–190.

[94] L. Xu, C. Lu, Y. Xu, and J. Jia, "Image smoothing via L_0 gradient minimization," in *Proc. SIGGRAPH Asia Conf.*, Dec. 2011, pp. 1–12.

[95] L. Xu, Q. Yan, Y. Xia, and J. Jia, "Structure extraction from texture via relative total variation," *ACM Trans. Graph.*, vol. 31, no. 6, pp. 1–10, Nov. 2012.

[96] Q. Guo, X. Li, F. Juefei-Xu, H. Yu, Y. Liu, and S. Wang, "JPGNet: Joint predictive filtering and generative network for image inpainting," in *Proc. 29th ACM Int. Conf. Multimedia*, vol. 33, Oct. 2021, pp. 386–394.

[97] H. Wu, J. Zhou, and Y. Li, "Deep generative model for image inpainting with local binary pattern learning and spatial attention," *IEEE Trans. Multimedia*, vol. 24, pp. 4016–4027, 2022.

[98] M. Pietikäinen, "Local binary patterns," *Scholarpedia*, vol. 5, no. 3, p. 9775, 2010.

[99] F. Li, A. Li, J. Qin, H. Bai, W. Lin, R. Cong, and Y. Zhao, "SRInpaintor: When super-resolution meets transformer for image inpainting," *IEEE Trans. Comput. Imag.*, vol. 8, pp. 743–758, 2022.

[100] M. Huang, W. Yu, and L. Zhang, "DF3Net: Dual frequency feature fusion network with hierarchical transformer for image inpainting," *Inf. Fusion*, vol. 111, Nov. 2024, Art. no. 102487.

[101] Y. Zeng, Z. Lin, H. Lu, and V. M. Patel, "CR-fill: Generative image inpainting with auxiliary contextual reconstruction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14144–14153.

[102] R. Xu, M. Guo, J. Wang, X. Li, B. Zhou, and C. C. Loy, "Texture memory-augmented deep patch-based image inpainting," *IEEE Trans. Image Process.*, vol. 30, pp. 9112–9124, 2021.

[103] W. Quan, R. Zhang, Y. Zhang, Z. Li, J. Wang, and D.-M. Yan, "Image inpainting with local and global refinement," *IEEE Trans. Image Process.*, vol. 31, pp. 2405–2420, 2022.

[104] Y. Chen, R. Xia, K. Zou, and K. Yang, "FFTI: Image inpainting algorithm via features fusion and two-steps inpainting," *J. Vis. Commun. Image Represent.*, vol. 91, Mar. 2023, Art. no. 103776.

[105] S. Qu, Z. Niu, J. Zhu, B. Dong, and K. Huang, "Structure first detail next: Image inpainting with pyramid generator," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2023, pp. 1265–1270.

[106] L. Liao, J. Xiao, Z. Wang, C.-W. Lin, and S. Satoh, "Guidance and evaluation: Semantic-aware image inpainting for mixed scenes," in *Proc. 16th Eur. Conf.*, Glasgow, U.K. Springer, Aug. 2020, pp. 683–700.

[107] H. Liu, B. Jiang, Y. Xiao, and C. Yang, "Coherent semantic attention for image inpainting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4169–4178.

[108] P. Ardino, Y. Liu, E. Ricci, B. Lepri, and M. de Nadai, "Semantic-guided inpainting network for complex urban scenes manipulation," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 9280–9287.

[109] J. Qiu, Y. Gao, and M. Shen, "Semantic-SCA: Semantic structure image inpainting with the spatial-channel attention," *IEEE Access*, vol. 9, pp. 12997–13008, 2021.

[110] M. Xie, M. Xia, X. Liu, C. Li, and T.-T. Wong, "Seamless Manga inpainting with semantics awareness," *ACM Trans. Graph.*, vol. 40, no. 4, pp. 1–11, Aug. 2021.

[111] T. Wang, H. Ouyang, and Q. Chen, "Image inpainting with external-internal learning and monochromic bottleneck," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 5116–5125.

[112] A. Lahiri, A. K. Jain, S. Agrawal, P. Mitra, and P. K. Biswas, "Prior guided GAN based semantic inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13693–13702.

[113] L. Zhang, C. Barnes, K. Wampler, S. Amirghodsi, E. Shechtman, Z. Lin, and J. Shi, "Inpainting at modern camera resolution by guided patchmatch with auto-curation," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2022, pp. 51–67.

[114] S. Y. Kim, K. Aberman, N. Kanazawa, R. Garg, N. Wadhwa, H. Chang, N. Karnad, M. Kim, and O. Liba, "Zoom-to-inpaint: Image inpainting with high-frequency details," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 476–486.

[115] Y. Wang, Y.-C. Chen, X. Tao, and J. Jia, "VCNet: A robust approach to blind image inpainting," in *Proc. 16th Eur. Conf.*, Glasgow, U.K. Springer, 2020, pp. 752–768.

[116] Q. Sun, L. Ma, S. Joon Oh, L. V. Gool, B. Schiele, and M. Fritz, "Natural and effective obfuscation by head inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5050–5059.

[117] Y. Poirier-Ginter and J.-F. Lalonde, "Robust unsupervised StyleGAN image restoration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 22292–22301.

[118] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of StyleGAN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8107–8116.

[119] H. Liu, Z. Wan, W. Huang, Y. Song, X. Han, and J. Liao, "PD-GAN: Probabilistic diverse GAN for image inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9367–9376.

[120] Y. Zeng, Y. Gong, and J. Zhang, "Feature learning and patch matching for diverse image inpainting," *Pattern Recognit.*, vol. 119, Nov. 2021, Art. no. 108036.

[121] S. Zhao, J. Cui, Y. Sheng, Y. Dong, X. Liang, E. I. Chang, and Y. Xu, "Large scale image completion via co-modulated generative adversarial networks," 2021, *arXiv:2103.10428*.

[122] H. Zheng, Z. Lin, J. Lu, S. Cohen, E. Shechtman, C. Barnes, J. Zhang, N. Xu, S. Amirghodsi, and J. Luo, "Image inpainting with cascaded modulation GAN and object-aware training," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2022, pp. 277–296.

[123] A. B. Yildirim, H. Pehlivan, B. B. Bilecen, and A. Dundar, "Diverse inpainting and editing with GAN inversion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, vol. 33, Oct. 2023, pp. 23063–23073.

[124] D. P Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[125] C.-T. Tu and Y.-F. Chen, "Facial image inpainting with variational autoencoder," in *Proc. 2nd Int. Conf. Intell. Robotic Control Eng. (IRCE)*, Aug. 2019, pp. 119–122.

[126] X. Han, Z. Wu, W. Huang, M. Scott, and L. Davis, "FiNet: Compatible and diverse fashion image inpainting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4480–4490.

[127] A. Razavi, A. Van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with VQ-VAE-2," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–11.

[128] C. Zheng, T.-J. Cham, and J. Cai, "Pluralistic image completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1438–1447.

[129] C. Zheng, T.-J. Cham, and J. Cai, "Pluralistic free-form image completion," *Int. J. Comput. Vis.*, vol. 129, no. 10, pp. 2786–2805, Oct. 2021.

[130] J. Peng, D. Liu, S. Xu, and H. Li, "Generating diverse structure for image inpainting with hierarchical VQ-VAE," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10770–10779.

[131] Q. Liu, Z. Tan, D. Chen, Q. Chu, X. Dai, Y. Chen, M. Liu, L. Yuan, and N. Yu, "Reduce information loss in transformers for pluralistic image inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11337–11347.

[132] Y. Yu, F. Zhan, R. Wu, J. Pan, K. Cui, S. Lu, F. Ma, X. Xie, and C. Miao, "Diverse image inpainting with bidirectional and autoregressive transformers," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 69–78.

[133] Z. Wan, J. Zhang, D. Chen, and J. Liao, "High-fidelity pluralistic image completion with transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 4672–4681.

[134] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9992–10002.

[135] W. Li, Z. Lin, K. Zhou, L. Qi, Y. Wang, and J. Jia, "MAT: Mask-aware transformer for large hole image inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 10748–10758.

[136] J. L. F. Campana, L. G. L. Decker, M. R. E. Souza, H. D. A. Maia, and H. Pedrini, "Variable-hyperparameter visual transformer for efficient image inpainting," *Comput. Graph.*, vol. 113, pp. 57–68, Jun. 2023.

[137] J. Liu, M. Gong, Y. Gao, Y. Lu, and H. Li, "Bidirectional interaction of CNN and transformer for image inpainting," *Knowl.-Based Syst.*, vol. 299, Sep. 2024, Art. no. 112046.

[138] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool, "RePaint: Inpainting using denoising diffusion probabilistic models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11451–11461.

[139] S. Xie, Z. Zhang, Z. Lin, T. Hinz, and K. Zhang, "SmartBrush: Text and shape guided object inpainting with diffusion model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 22428–22437.

[140] P. Esser, R. Rombach, A. Blattmann, and B. Ommer, "ImageBART: Bidirectional context with multinomial diffusion for autoregressive image synthesis," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 3518–3532.

[141] E. Hoogeboom, D. Nielsen, P. Jaini, P. Forre, and M. Welling, "Argmax flows and multinomial diffusion: Learning categorical distributions," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 12454–12465.

[142] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. Van Den Berg, "Structured denoising diffusion models in discrete state-spaces," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 17981–17993.

[143] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 10674–10685.

[144] H. Chung, B. Sim, and J. C. Ye, "Come-closer-diffuse-faster: Accelerating conditional diffusion models for inverse problems through stochastic contraction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 12403–12412.

[145] P. Peter, K. Schrader, T. Alt, and J. Weickert, "Deep spatial and tonal data optimisation for homogeneous diffusion inpainting," *Pattern Anal. Appl.*, vol. 26, no. 4, pp. 1585–1600, Nov. 2023.

[146] B. Fei, Z. Lyu, L. Pan, J. Zhang, W. Yang, T. Luo, B. Zhang, and B. Dai, "Generative diffusion prior for unified image restoration and enhancement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 9935–9946.

[147] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," *Robotica*, vol. 17, no. 2, pp. 229–235, 1999.

[148] H. Wang, Y. Tang, Y. Wang, J. Guo, Z.-H. Deng, and K. Han, "Masked image modeling with local multi-scale reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 2122–2131.

[149] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, Nonlinear Phenomena*, vol. 60, nos. 1–4, pp. 259–268, Nov. 1992.

[150] A. Lahiri, A. K. Jain, D. Nadendla, and P. K. Biswas, "Faster unsupervised semantic inpainting: A GAN based approach," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 2706–2710.

[151] O. Losson, L. Macaire, and Y. Yang, "Comparison of color demosaicing methods," in *Advances in Imaging and Electron Physics*, vol. 162. Amsterdam, The Netherlands: Elsevier, 2010, pp. 173–265.

[152] C. Haccius and T. Herfet, "Computer vision performance and image quality metrics: Areciprocal relation," *Comput. Vis. Perform. Image Quality Metrics Reciprocal Relation*, vol. 1, pp. 27–37, Jan. 2017.

[153] B. Sankur, "Statistical evaluation of image quality measures," *J. Electron. Imag.*, vol. 11, no. 2, pp. 206–223, Apr. 2002.

[154] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[155] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. 37th Asilomar Conf. Signals, Syst. Comput.*, 2003, pp. 1398–1402.

[156] Y. Yu, F. Zhan, S. Lu, J. Pan, F. Ma, X. Xie, and C. Miao, "WaveFill: A wavelet-based generation network for image inpainting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14094–14103.

[157] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5967–5976.

[158] L. Jiang, B. Dai, W. Wu, and C. C. Loy, "Focal frequency loss for image reconstruction and synthesis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 13899–13909.

[159] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 1–9.

[160] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–12.

[161] Y. Alami Mejjati, C. Richardt, J. Tompkin, D. Cosker, and K. I. Kim, "Unsupervised attention-guided image-to-image translation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–11.

[162] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 586–595.

[163] I. Goodfellow, "NIPS 2016 tutorial: Generative adversarial networks," 2017, *arXiv:1701.00160*.

[164] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[165] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4396–4405.

[166] R. Zhang, W. Quan, B. Wu, Z. Li, and D. Yan, "Pixel-wise dense detector for image inpainting," *Comput. Graph. Forum*, vol. 39, no. 7, pp. 471–482, Oct. 2020.

[167] J. You and J. Korhonen, "Transformer for image quality assessment," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 1389–1393.

[168] M. Cheon, S.-J. Yoon, B. Kang, and J. Lee, "Perceptual image quality assessment with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 433–442.

[169] J. Ke, Q. Wang, Y. Wang, P. Milanfar, and F. Yang, "MUSIQ: Multi-scale image quality transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 5128–5137.

[170] K. Iskakov, "Semi-parametric image inpainting," 2018, *arXiv:1807.02855*.

[171] E. Agustsson and R. Timofte, "NTIRE 2017 challenge on single image super-resolution: Dataset and study," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1122–1131.

[172] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, and M. Wortsman, "LAION-5B: An open large-scale dataset for training next generation image-text models," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 25278–25294.

[173] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros, "What makes Paris look like paris?" *Commun. ACM*, vol. 58, no. 12, pp. 103–110, Nov. 2015.

[174] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[175] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. 13th Eur. Conf.*, Zurich, Switzerland. Springer, Sep. 2014, pp. 740–755.

[176] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1452–1464, Jun. 2018.

[177] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3730–3738.

[178] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," 2017, *arXiv:1710.10196*.

[179] B. M. Smith, L. Zhang, J. Brandt, Z. Lin, and J. Yang, "Exemplar-based face parsing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3484–3491.

[180] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang, "Interactive facial feature localization," in *Proc. 12th Eur. Conf. Comput. Vis.*, Florence, Italy. Springer, Oct. 2012, pp. 679–692.

[181] J. Liu and C. Jung, "Facial image inpainting using attention-based multi-level generative network," *Neurocomputing*, vol. 437, pp. 95–106, May 2021.

[182] J. Li, Z. Li, J. Cao, X. Song, and R. He, "FaceInpainter: High fidelity face adaptation to heterogeneous domains," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 5085–5094.

[183] H. Li, W. Wang, C. Yu, and S. Zhang, "SwapInpaint: Identity-specific face inpainting with identity swapping," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 7, pp. 4271–4281, Jul. 2022.

[184] X. Zhang, X. Wang, C. Shi, Z. Yan, X. Li, B. Kong, S. Lyu, B. Zhu, J. Lv, Y. Yin, Q. Song, X. Wu, and I. Mumtaz, "DE-GAN: Domain embedded GAN for high quality face image inpainting," *Pattern Recognit.*, vol. 124, Apr. 2022, Art. no. 108415.

[185] Y. Jiang, F. Yang, Z. Bian, C. Lu, and S. Xia, "Mask removal: Face inpainting via attributes," *Multimedia Tools Appl.*, vol. 81, no. 21, pp. 29785–29797, Sep. 2022.

[186] J. Yu, K. Li, and J. Peng, "Reference-guided face inpainting with reference attention network," *Neural Comput. Appl.*, vol. 34, no. 12, pp. 9717–9731, Jun. 2022.

[187] A. Hassanpour, A. E. Daryani, M. Mirmahdi, K. Raja, B. Yang, C. Busch, and J. Fierrez, "E2F-GAN: Eyes-to-face inpainting via edge-aware coarse-to-fine GANs," *IEEE Access*, vol. 10, pp. 32406–32417, 2022.

[188] L. He, Z. Qiang, X. Shao, H. Lin, M. Wang, and F. Dai, "Research on high-resolution face image inpainting method based on StyleGAN," *Electronics*, vol. 11, no. 10, p. 1620, May 2022.

[189] J. Xu, S. Motamed, P. Vaddamanu, C. H. Wu, C. Haene, J.-C. Bazin, and F. De La Torre, "Personalized face inpainting with diffusion models by parallel visual attention," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, vol. 33, Jan. 2024, pp. 5420–5430.

[190] X. Ma, X. Zhou, H. Huang, G. Jia, Z. Chai, and X. Wei, "Contrastive attention network with dense field estimation for face completion," *Pattern Recognit.*, vol. 124, Apr. 2022, Art. no. 108465.

[191] E. Upenik, P. Akyazi, M. Tuzmen, and T. Ebrahimi, "Inpainting in omnidirectional images for privacy protection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 2487–2491.

[192] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS*, 2011, p. 4.

[193] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D object representations for fine-grained categorization," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Dec. 2013, pp. 554–561.

[194] R. Tyleček and R. Šára, "Spatial pattern templates for recognition of objects with regular structure," in *Proc. 35th German Conf.*, Saarbrücken, Germany. Springer, Sep. 2013, pp. 364–374.

[195] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing textures in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3606–3613.

[196] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic scene completion from a single depth image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 190–198.

[197] A. Grigorev, A. Sevastopolsky, A. Vakhitov, and V. Lempitsky, "Coordinate-based texture inpainting for pose-guided human image generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12127–12136.

[198] A. Cui, D. McKee, and S. Lazebnik, "Dressing in order: Recurrent person image generation for pose transfer, virtual try-on and outfit editing," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14618–14627.

[199] Z. Wan, B. Zhang, D. Chen, P. Zhang, D. Chen, J. Liao, and F. Wen, "Bringing old photos back to life," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2744–2754.

[200] H. Chen, M. V. Giuffrida, P. Doerner, and S. A. Tsaftaris, "Adversarial large-scale root gap inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 2619–2628.

[201] Y.-H. Cheng and P.-Y. Chen, "Using generative adversarial network technology for repairing dynamically blurred license plates," in *Proc. 6th Int. Symp. Comput., Consum. Control (ISC)*, Jun. 2023, pp. 126–129.

[202] R. Wan, B. Shi, H. Li, Y. Hong, L.-Y. Duan, and A. C. Kot, "Benchmarking single-image reflection removal algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 2, pp. 1424–1441, Feb. 2023.

[203] K. Armanious, Y. Mecky, S. Gatidis, and B. Yang, "Adversarial inpainting of medical image modalities," in *Proc. ICASSP - IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 3267–3271.

[204] Y. Chai, B. Xu, K. Zhang, N. Lepore, and J. C. Wood, "MRI restoration using edge-guided adversarial learning," *IEEE Access*, vol. 8, pp. 83858–83870, 2020.

[205] Y. Miao, Y. Sun, S. Li, P. Zhang, Y. Yang, and Y. Hu, "Spinal neoplasm image inpainting with deep convolutional neutral networks," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2019, pp. 2619–2624.

[206] Q. Wang, Y. Chen, N. Zhang, and Y. Gu, "Medical image inpainting with edge and structure priors," *Measurement*, vol. 185, Nov. 2021, Art. no. 110027.

[207] N. V. Gapon, V. V. Voronin, R. A. Sizyakin, D. Bakaev, and A. Skorikova, "Medical image inpainting using multi-scale patches and neural networks concepts," *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 680, no. 1, Nov. 2019, Art. no. 012040.

[208] K. Xie, L. Gao, H. Zhang, S. Zhang, Q. Xi, F. Zhang, J. Sun, T. Lin, J. Sui, and X. Ni, "Inpainting truncated areas of CT images based on generative adversarial networks with gated convolution for radiotherapy," *Med. Biol. Eng. Comput.*, vol. 61, no. 7, pp. 1757–1772, Jul. 2023.

[209] L. Jiao, Z. Huang, X. Lu, X. Liu, Y. Yang, J. Zhao, J. Zhang, B. Hou, S. Yang, F. Liu, and W. Ma, "Brain-inspired remote sensing foundation models and open problems: A comprehensive survey," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 10084–10120, 2023.

[210] M. Shao, C. Wang, T. Wu, D. Meng, and J. Luo, "Context-based multiscale unified network for missing data reconstruction in remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2020.

[211] H. Xu, X. Tang, B. Ai, X. Gao, F. Yang, and Z. Wen, "Missing data reconstruction in VHR images based on progressive structure prediction and texture generation," *ISPRS J. Photogramm. Remote Sens.*, vol. 171, pp. 266–277, Jan. 2021.

[212] J. Li, Z. Wu, Z. Hu, J. Zhang, M. Li, L. Mo, and M. Molinier, "Thin cloud removal in optical remote sensing images based on generative adversarial networks and physical model of cloud distortion," *ISPRS J. Photogramm. Remote Sens.*, vol. 166, pp. 373–389, Aug. 2020.

[213] Y. Zi, F. Xie, N. Zhang, Z. Jiang, W. Zhu, and H. Zhang, "Thin cloud removal for multispectral remote sensing images using convolutional neural networks combined with an imaging model," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 3811–3823, 2021.

[214] H. Pan, "Cloud removal for remote sensing imagery via spatial attention generative adversarial network," 2020, *arXiv:2009.13015*.

[215] F. N. Darbaghshahi, M. R. Mohammadi, and M. Soryani, "Cloud removal in remote sensing images using generative adversarial networks and SAR-to-optical image translation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4105309.

[216] A. Habib, P. Banerjee, S. Mishra, N. Yadav, K. Agarwal, F. Melandsø, and D. K. Prasad, "Image inpainting in acoustic microscopy," Tech. Rep., 2023.

[217] Y. Wu, R. Deng, S. Linghu, J. Dong, and Y. Yang, "Method of image restoration of the blank strips of electric imaging logs," *Arabian J. Geosci.*, vol. 15, no. 13, p. 1189, Jul. 2022.

[218] T. Yu, C. Lin, S. Zhang, S. You, X. Ding, J. Wu, and J. Zhang, "End-to-end partial convolutions neural networks for Dunhuang Grottoes wall-painting restoration," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 1447–1455.

[219] J. Cao, Z. Zhang, A. Zhao, H. Cui, and Q. Zhang, "Ancient mural restoration based on a modified generative adversarial network," *Heritage Sci.*, vol. 8, no. 1, pp. 1–14, Dec. 2020.

[220] I.-M. Ciortan, S. George, and J. Y. Hardeberg, "Colour-balanced edge-guided digital inpainting: Applications on artworks," *Sensors*, vol. 21, no. 6, p. 2091, Mar. 2021.
[221] N. Wang, W. Wang, W. Hu, A. Fenster, and S. Li, "Thanka mural inpainting based on multi-scale adaptive partial convolution and stroke-like mask," *IEEE Trans. Image Process.*, vol. 30, pp. 3720–3733, 2021.
[222] A. Bar, Y. Gandelsman, T. Darrell, A. Globerson, and A. Efros, "Visual prompting via image inpainting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 25005–25017.
[223] T. Yu, R. Feng, R. Feng, J. Liu, X. Jin, W. Zeng, and Z. Chen, "Inpaint anything: Segment anything meets image inpainting," 2023, *arXiv:2304.06790*.
[224] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, and W.-Y. Lo, "Segment anything," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2023, pp. 4015–4026.
[225] S. Xu, X. Wang, W. Lv, Q. Chang, C. Cui, K. Deng, G. Wang, Q. Dang, S. Wei, Y. Du, and B. Lai, "PP-YOLOE: An evolved version of Yolo," 2022, *arXiv:2203.16250*.
[226] R. Thoppilan, D. De Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H.-T. Cheng, A. Jin, T. Bos, L. Baker, and Y. Du, "LaMDA: Language models for dialog applications," 2022, *arXiv:2201.08239*.
[227] J. Zhuang, Y. Zeng, W. Liu, C. Yuan, and K. Chen, "A task is worth one word: Learning with task prompts for high-quality versatile image inpainting," 2023, *arXiv:2312.03594*.
[228] C. Wu, S. Yin, W. Qi, X. Wang, Z. Tang, and N. Duan, "Visual ChatGPT: Talking, drawing and editing with visual foundation models," 2023, *arXiv:2303.04671*.
[229] S. Jain, V. Shivam, A. P. Bidargaddi, S. Malipatil, and K. Patil, "Image inpainting using YOLOv8 and Lama model," in *Proc. 5th Int. Conf. Emerg. Technol. (INCET)*, vol. 51, May 2024, pp. 1–7.
[230] X. Yue, K. Qi, X. Na, Y. Zhang, Y. Liu, and C. Liu, "Improved YOLOv8-Seg network for instance segmentation of healthy and diseased tomato plants in the growth stage," *Agriculture*, vol. 13, no. 8, p. 1643, Aug. 2023.
[231] R. Bodur, E. Gundogdu, B. Bhattarai, T.-K. Kim, M. Donoser, and L. Bazzani, "iEdit: Localised text-guided image editing with weak supervision," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2024, pp. 7426–7435.
[232] J. Su, B. Xu, and H. Yin, "A survey of deep learning approaches to image restoration," *Neurocomputing*, vol. 487, pp. 46–65, May 2022.

JING YANG received the B.Sc. degree in software engineering from Shandong University, China, and the M.Sc. degree in software technology from The Hong Kong Polytechnic University. She is currently pursuing the Ph.D. degree with the School of Computer Sciences, Universiti Sains Malaysia. She is also with Shanxi Datong University, China. Her research interests include the application of computer vision to the protection of cultural relics.

NUR INTAN RAIHANA RUHAIYEM received the Ph.D. degree in computational cell biology from The University of Queensland, Australia, in 2014. She is currently a Senior Lecturer with the School of Computer Sciences, Universiti Sains Malaysia. She is also a Certified Trainer with Human Resources Development Corporation (HRD Corp), Malaysia, and has handled many workshops in the field of data science specifically in data visualization and analytics, since 2018. Her main research interests and publications are in the areas of medical image processing and analysis, computer vision, and data visualization and analysis.