Information Fusion 104 (2024) 102174

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Structure-aware multi-view image inpainting using dual consistency attention

Hongyue Xiang [a], Weidong Min [a,b,c,*], Qing Han [a,b,c], Cheng Zha [a], Qian Liu [a], Meng Zhu [a]

[a] School of Mathematics and Computer Science, Nanchang University, Nanchang, 330031, China
[b] Institute of Metaverse, Nanchang University, Nanchang, 330031, China
[c] Jiangxi Key Laboratory of Smart City, Nanchang, 330031, China

ARTICLE INFO

Keywords:
Image inpainting
Multi-view
Structure-aware
Dual consistency attention
Image local refinement

ABSTRACT

Image inpainting based on deep learning has made remarkable progress and is widely used in image editing, cultural relic preservation, etc. However, most image inpainting methods are implemented based on single-view images. This does not fully utilize the known information and leads to unsatisfactory inpainting results. Moreover, these methods usually ignore the importance of image consistency and the surrounding regions, leading to irrelevant contents and visual artifacts in the inpainting results. To solve these problems, a structure-aware multi-view image inpainting method using dual consistency attention (SM-DCA) is proposed in this paper. It consists of two parts. The first part is the structure-aware multi-view image inpainting. This part constructs structure views as additional views to assist image inpainting. It is implemented by two networks: a structure inpainting network with strong constraints (SSC) and an image inpainting network with dual consistency attention (IDCA). SSC is used to repair structure views and make them closely resemble the ground truth through strong constraints. IDCA improves the consistency between the generated content and the whole image, making the repaired image more reasonable. The second part is image refinement, implemented by an image local refinement network (ILR). It can focus on the surrounding regions, eliminating boundary artifacts and obtaining finer local details. In Paris StreetView, SM-DCA achieves 22.0194, 0.7457 and 0.0557 in terms of PSNR, SSIM and MAE at 50%–60% damage. The corresponding values in CelebA are 22.5526, 0.8623 and 0.0453, respectively. The extensive experimental results on the Paris StreetView and CelebA datasets demonstrate the superiority of SM-DCA.

1. Introduction

As a research hot spot in computer vision, image inpainting refers to reasonably inferring the content of unknown regions based on known information [1]. The goal of image inpainting methods is that the inferred contents should be consistent with the known contents in terms of color, texture, structure, and semantics. These techniques for inferring unknown image information are used in many fields, such as image editing, image super-resolution, image denoising, removal of unwanted objects, etc [2–7].

During the decades of image inpainting development, lots of methods have been proposed. Early works can be divided into two categories: patch-based and diffusion-based methods [6–9]. Patch-based methods fill in the missing contents by searching for the most similar patches based on similarity computations. Diffusion-based methods use partial differential equations and variation methods to diffuse the surrounding information into the interior of the missing regions, completing the content filling. These methods are called traditional methods. They can achieve a good inpainting effect when the missing regions are small. However, they fail to generate novel content and coherent semantic information, so it is difficult for traditional methods to handle large missing regions.

Deep learning technology has been widely used in many fields due to its unique characteristics, including object detection, vehicle re-identification, image inpainting, etc [10–16]. Image inpainting methods based on deep learning can generate new relevant contents depending on the known contents and obtain plausible inpainting results. These methods typically use single-view images as input. They can be easily implemented at a low computational cost. However, it is difficult for them to extract comprehensive features and take full advantage of the known information, negatively affecting inpainting results. To solve this problem, the multi-view learning has been applied into image inpainting [17]. As an information fusion method, the methods based on multi-view learning can extract more comprehensive and robust

* Corresponding author at: School of Mathematics and Computer Science, Nanchang University, Nanchang, 330031, China.
E-mail address: minweidong@ncu.edu.cn (W. Min).

https://doi.org/10.1016/j.inffus.2023.102174
Received 1 September 2023; Received in revised form 19 November 2023; Accepted 30 November 2023
Available online 1 December 2023

features. They construct additional views based on the original ones to assist image inpainting and improve the inpainting performance. There are many forms in the additional views, such as edges [18–20], contours [21], multi-resolution features [22], etc. The use of additional views can improve the performance of image inpainting, especially for large missing cases with less known information. However, this also poses a problem. Errors in the repaired additional view would be transmitted to the original view inpainting, resulting in distorted structures, blurred textures, falsely filled contents, etc. Thus, it is important to ensure the accuracy of the repaired additional views.

The objective of image inpainting is to fill in missing regions and make the generated contents visually consistent with the known contents. The generated contents of a good image inpainting method should have high consistency with the known contents. To achieve this objective and further improve the inpainting performance, attention mechanisms have been introduced and are widely used [23–27]. They assign different weights to different parts of an image and enable the inpainting network to focus on useful regions, ensuring the consistency of the repaired image. However, these mechanisms usually only consider the consistency between the generated contents and the known contents, ignoring the internal consistency within the generated contents. This is not conducive to generating high-consistency inpainting images. To generate plausible repaired images, the internal consistency within the generated contents is also crucial.

Another reason for unsatisfactory inpainting results is the overpursuit of large receptive fields, ignoring the influence of information in the surrounding region. There are many types of damage. Not every type requires a large receptive field to complete the inpainting, and some damage types are simply related to the surrounding regions. For example, in the case of narrow damage, the missing content is highly correlated with information from the surrounding region rather than distant information. Thus, to avoid apparent boundaries and obtain plausible inpainting results with fine details, it is necessary to take into account information from the surrounding region.

In summary, the reasons why current image inpainting methods struggle to generate satisfactory and reasonable results can be attributed to the following three points: 1. Using single-view inpainting that fails to fully utilize known information. 2. Considering consistency between inpainting and known content but neglecting internal consistency among inpainting content. 3. Excessive pursuit of large receptive fields, overlooking the influence of surrounding regions on the inpainting process. To address these issues, we propose a novel structure-aware multi-view image inpainting method based on dual consistency attention (SM-DCA). It can be divided into two parts: structure-aware multi-view image inpainting and image refinement. The former part consists of the structure inpainting network with strong constraints (SSC) and the image inpainting network based on dual consistency attention (IDCA). SSC is designed to extract structure views as the additional views and repair the structure views. Its strong constraints enable the repaired structure views to closely resemble the ground truth. IDCA uses the previously repaired structure views as guidance to assist image inpainting, where dual consistency attention (DCA) is proposed to improve the internal and external consistency. The internal consistency refers to the consistency among inpainting content, and the external one is the consistency between inpainting and known content. The image refinement part is implemented by the image local refinement network with a small receptive field (ILR). It is used to consider the surrounding regions and further refine the inpainting images, eliminating apparent boundaries and generating finer details. The contributions of our proposed method can be summarized as follows.

(1) Construct structure views as additional views to guide image inpainting to achieve multi-view image inpainting for satisfactory inpainting results. SSC is designed to ensure the accuracy of repaired structure views.

(2) DCA is proposed to generate inpainting results with high consistency. It considers internal consistency and external consistency simultaneously based on similarity scores.

(3) ILR is designed to refine the previous results. It focuses on the surrounding regions based on small receptive fields, eliminating boundary artifacts, and generating fine local details.

The rest of the paper is organized as follows. Related works is discussed in Section 2. An overview of SM-DCA is presented in Section 3. In Section 4, the three networks of SM-DCA are separately described in detail. Section 5 shows the quantitative and qualitative experimental results, and the corresponding analysis is also described in this section. In addition, ablation experimental results and analysis are also shown in this section. Section 6 concludes our work.

2. Related work

Existing methods for image inpainting are usually divided into two categories: traditional methods and deep learning methods. Traditional image inpainting methods can be broadly divided into two categories: patch-based and diffusion-based methods. Patch-based methods fill in missing regions by searching for similar patches in known regions. One of the classical methods is that proposed by Criminis et al. [6]. But it takes too much time to search for the most similar patches because the selected patches need to be compared with all the patches in the image. Then Barnes et al. [7] propose the PatchMatch method, where a randomized nearest neighbor patch matching algorithm is designed to solve this problem. Diffusion-based methods typically start from the holes' boundary and iteratively diffuse inward based on the surrounding information. The diffusion technique is first applied to image inpainting by Bertalmio et al. [8]. They complete the filling from outside-in and thick-to-thin, and propagate in the direction of an estimated isophote by simulating the process of manually patching artworks. Chan and Shen [9] then propose a curvature-driven diffusion method for repairing non-textured images, which can better propagate edge information. These traditional image inpainting methods can achieve good performance when dealing with small damage regions. But they struggle to deal with large damage regions due to their inability to understand image semantics and generate novel contents.

Deep learning techniques are widely used in image inpainting due to their powerful learning capabilities. In early works, Pathak et al. [13] propose an image inpainting method based on encoder–decoder architecture (CE) that maps a corrupted image to a complete image. However, its adversarial loss is applied only to the missing regions and not the whole image, which leads to blurring and low image consistency. To solve this problem, Iizuka et al. [14] design a novel network that introduces a global discriminator and a local discriminator to enhance global and local consistency. But the local discriminator is more suitable for rectangular holes. To deal with irregular holes inpainting, Liu et al. [15] design a partial convolution operation and an automatic mask-update mechanism. The defect of this method is that the mask would gradually disappear as the number of network layers increased. These methods only use single-view images as input. This makes them easy to implement but difficult to generate satisfactory results in complex scenarios due to the lack of sufficient constraints.

To improve image inpainting performance, image inpainting methods based on multi-view learning algorithms have been proposed. Multi-view learning algorithms synthesize data or features obtained by observing an object from multiple perspectives to judge what the object really is [28–30]. Multi-view data has begun to emerge in large numbers, which means that the same objects are described from different perspectives [31]. It consists of multi-source, multi-angle, multi-scale, and multi-feature data. As an information fusion method, multi-view learning can improve model performance by integrating information from multiple views to extract more accurate, semantically richer, and more comprehensive feature representations. EdgeConnect [18], proposed by Nazeri et al. uses edge images obtained by the canny operator as additional views. In E2I proposed by Xu et al. the extraction is implemented by Holistically-nested Edge Detection (HED) [19]. Xiong et al. use DeepCut as the extraction operation to predict a salient object mask

with accurate boundaries, named as contour image [21]. The contour images are used as additional views to guide the image inpainting. In Structure-Flow, proposed by Ren et al. [32], edge-preserved smooth images are employed as the global structure information to guide the second stage of inpainting. The appearance flow is then introduced to yield image details. The texture-aware multi-GAN proposed by Hedjazi and Genc trains four GAN networks to complete image inpainting [22]. It constructs four views with different resolutions and accordingly trains four GANs. Regardless of the network design, the fundamental idea of multi-view based image inpainting is to provide guidance on the subsequent inpainting process by constructing additional views. This is beneficial for improving inpainting performance. But this also poses a problem. Errors may be present in the repaired additional views, leading to errors in the final inpainting results. Thus, it is important to add sufficient constraints and ensure the accuracy of the repaired additional views.

Moreover, attention mechanisms have been widely used to improve model performance for image inpainting. Yu et al. propose a contextual attention module to model the long-term correlation between distant contextual information and the missing regions [23]. This module learns where to borrow or copy feature information from known background patches to generate missing patches. Zheng et al. design a novel short-long term context attention layer that exploits the distance relation between decoder and encoder features to improve appearance consistency [24]. PEN-Net is proposed by Zeng et al. to ensure visual and semantic coherence via a cross-layer attention transfer mechanism [25]. It learns region affinity from high-level feature maps and uses the learned affinity to guide the inpainting of the adjacent low-level layer. However, most attention mechanisms only consider the consistency between the generated contents and the known contents, ignoring the consistency within the generated contents. This leads to unsatisfactory inpainting results. In addition, deep learning methods always attempt to improve the image inpainting performance by increasing the network depth and enlarging the receptive field size. This operation disregards the importance of the surrounding regions in the inpainting process, potentially resulting in apparent color differences and undesirable details.

3. Overview

The overview of SM-DCA is shown in Fig. 1. Overall, SM-DCA consists of two parts: structure-aware multi-view image inpainting and image refinement. The former part is implemented by the structure inpainting network with strong constraints (SSC) and the image inpainting network based on dual consistency attention (IDCA). SSC is designed to extract and repair structure views, enabling the repaired structure views to closely resemble the real structure views via strong constraints. IDCA is proposed to complete the image inpainting with the guidance of the repaired structure views. In IDCA, the dual consistency attention module (DCA) is designed to ensure image consistency by considering internal and external consistency simultaneously. Here, the internal consistency refers to the consistency within the generated contents, while the external is the consistency between the generated content and the known contents. The image refinement part is designed to refine previously obtained results and is implemented by the image local refinement network (ILR). All three networks follow encoder–decoder architectures.

Specifically, assume that I_{gt} represents the ground truth images, I_m is the masked images, and M is the mask images. I_{gt}, I_m, and M are used as the input to SSC, where I_{gt} is only used in the training process. The real and masked structure views S_{gt} and S_m are constructed by the structure extraction, respectively. The values obtained from the real views are used to impose multi-view constraints on the masked views in SSC. This can make the generated features close to those of real views, improving the accuracy of the repaired structure views S_{out}. Then, S_{out}, I_m, and M are transmitted to IDCA to complete the image inpainting, and the output is denoted as I_{inter}. The design of DCA is beneficial for improving the quality and reasonability of the repaired images. ILR is then used to refine the repaired images I_{inter}. It uses a shallow network with a small receptive field, which allows the network to focus on the surrounding regions, generating finer local details and eliminating color inconsistencies. The output of ILR is denoted as I_l. Therefore, the final output of SM-DCA can be described as $I_{out} = I_l \odot M + I_m \odot (1 - M)$, which preserves the known contents unchanged. The details of these networks are provided in Section 4.

4. Method

The proposed method, SM-DCA, is constructed in two parts: structure-aware multi-view image inpainting and image refinement. The former part is implemented by two networks (SSC and IDCA); the latter is implemented by a shallow network (ILR). This section separately describes the three networks in detail.

4.1. Structure inpainting network with strong constraint

As the first step of the structure-aware multi-view image inpainting part, the structure inpainting network with strong constraint (SSC) is designed to construct and repair the structure views. The proposed strong constraints can enable the repaired structure views to closely resemble the ground truth views, avoiding adverse effects on the subsequent inpainting process. The SSC follows an adversarial model consisting of a generator G_s based on U-Net and a discriminator D_s based on the PatchGAN [33]. Its specific architecture is shown in Fig. 2. Here, the choice of U-Net as the backbone network is motivated by its remarkable ability to capture global contextual information and local details. Moreover, its skip connections allow the information from the encoding phase to be utilized in the decoding phase, thereby aiding the enhancement of image inpainting quality. In this figure, the black ellipsis dots represent the skip connections that were not drawn due to lack of space. Similarly, the green ellipsis dots denote the constraints.

In SSC, the input consists of the ground truth image I_{gt}, the masked image I_m ($I_m = I_{gt} \odot (1 - M)$) and the corresponding mask image M (where 1 indicates the missing region while 0 indicates the known region). I_{gt} is only used in the training process. During training, the real structure views S_{gt} and the masked structure views S_m are obtained by the structure extraction, respectively. The structure extraction operation is implemented by the modified canny operation. In the modified canny operation, a Boolean matrix is generated based on the mask images M, where the values covered by the missing regions are set to 0 (False) and the remaining regions are set to 1 (True). By this matrix, the boundaries of the mask regions can be eliminated and only the structure information covered by the known regions is maintained. The output of this network is the repaired structure views S_{out}, which is employed as a guide to assist the subsequent inpainting. The accuracy of S_{out} directly affects the accuracy of the entire image inpainting. To ensure the accuracy of the repaired structure views, the ground truth combination (I_{gt}, S_{gt}, M) and the masked combination (I_m, S_m, M) are separately fed into G_s. The features generated by the ground truth branch are employed to impose constraints on the features generated by the masked branch in each layer of G_s. The difference between the features of the two branches is computed to ensure that the generated features are close to the ground truth features. This is beneficial to improving the accuracy of the repaired structure images.

The strong constraints are implemented by the multi-level structure loss. It calculates the difference within the features generated by the masked and the ground truth branch. Assume that $F_{s_i}()$ denotes the feature generated by the ith layer of G_s. The differences of each layer are added together to obtain the final multi-level structure loss, which can be defined as Eq. (1).

$$L_{mul} = \sum_{i \in N} \left\| F_{s_i}(S_m) - F_{s_i}(S_{gt}) \right\|_2 \tag{1}$$

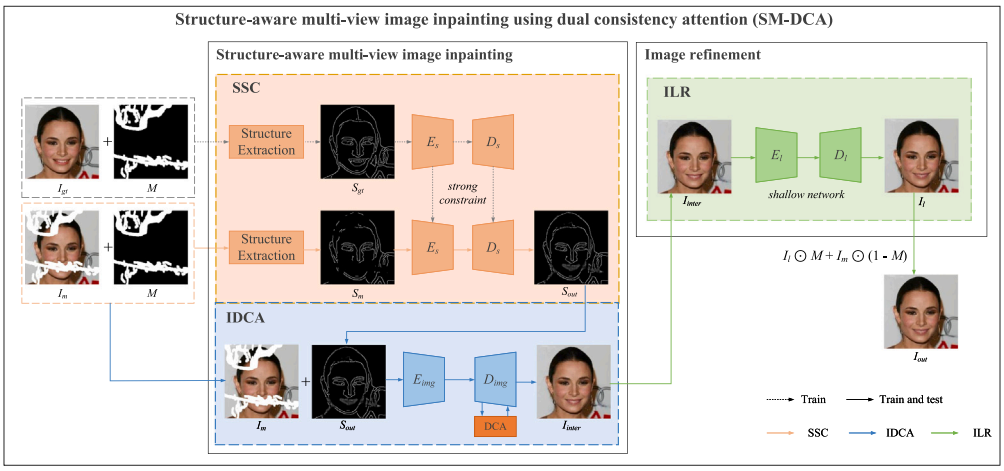

Fig. 1. The overview of SM-DCA.

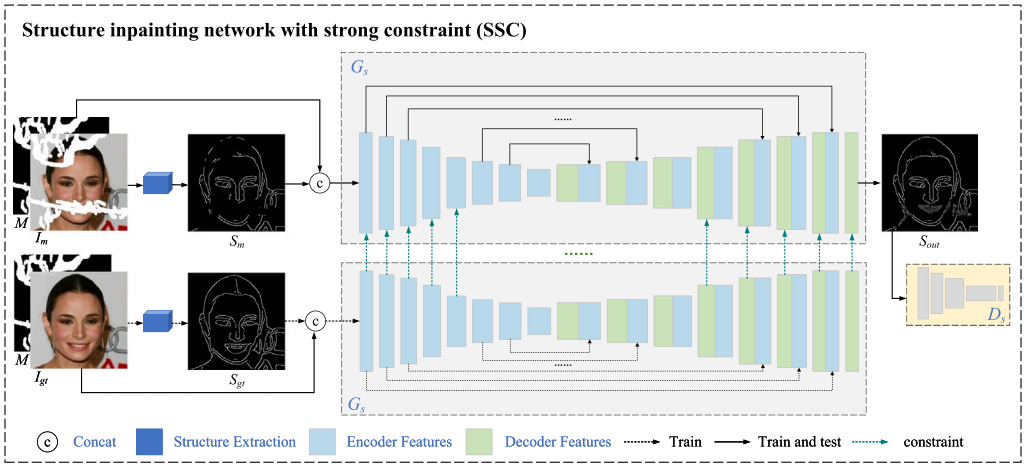

Fig. 2. The specific architecture of SSC.

where N represents the total layer number of G_s. By narrowing down the differences between the generated features and the ground truth features, the repaired structure views would be closer to the ground truth. This indicates that the accuracy of SSC could be improved.

In addition to the multi-level structure loss function, reconstruction loss, adversarial loss, and feature matching loss are also used to guarantee the accuracy of the repaired structure views in this network. The reconstruction loss is defined as Eq. (2), which computes the difference between the repaired views and the ground truth at the pixel level.

$$L_r = \left\| S_{out} - S_{gt} \right\|_2 \tag{2}$$

The adversarial loss is defined as Eq. (3). In this equation, the objective of G_s is minimizing the differences between the generated structure views and the ground truth. In contrast, the objective of D_s is maximizing these differences.

$$L_{adv} = \min_{G_s} \max_{D_s} \mathrm{E}_{S_{gt}} \left[\log D_s \left(S_{gt} \right) \right] + \mathrm{E}_{S_m} \left[\log \left(1 - D_s \left(G_s \left(S_m, M \right) \right) \right) \right] \tag{3}$$

The feature matching loss is expressed as Eq. (4).

$$L_{fm} = \mathrm{E} \left[\sum_i \frac{1}{N_i} \left\| D_s^{\,i} \left(S_{gt} \right) - D_s^{\,i} \left(S_{out} \right) \right\|_1 \right] \tag{4}$$

where D_s^i represents the ith layer of discriminator D_s. This indicates that the feature matching loss L_{fm} compares the output results of each intermediate layer within the discriminator D_s, improving the reality of the generated views and the stability of the GAN training.

In summary, the total loss of SSC can be indicated as Eq. (5).

$$L_{SSC} = \lambda_{mul} L_{mul} + \lambda_r L_r + \lambda_{adv} L_{adv} + \lambda_{fm} L_{fm} \tag{5}$$

where $\lambda_{mul} = \lambda_r = 1$, $\lambda_{adv} = 0.1$, and $\lambda_{fm} = 10$.

4.2. Image inpainting network with dual consistency attention

As the second step of the structure-aware multi-view image inpainting, the image inpainting network with dual consistency attention (IDCA) is described in this section. It is designed to complete image inpainting based on the guidance of the previously repaired structure views. Its specific architecture is shown in Fig. 3.

The input of IDCA is the combination of I_m, M, S_{out}, and the output is represented by I_{inter}. The figure indicates that the architecture is basically the same as that of the SSC, except that a dual consistency attention (DCA) module is embedded between the 12th and 13th layers. DCA is designed to take into account both internal and external consistency, where internal refers to the consistency within the generated contents, and external refers to the consistency between the generated contents and the known contents. The specific architecture of DCA is also shown in Fig. 3. Assume that f represents the input to DCA and m is the mask feature maps obtained by resizing the mask images M, where f and m have the same size. First, the $unfold(\cdot)$ function is taken to divide f and m into patches of size 3×3, obtaining $pf = unfold(f, 3)$ and $pm = unfold(m, 3)$. According to the mask patches pm, the feature patches pf can be divided into the generated contents (C_g) and the

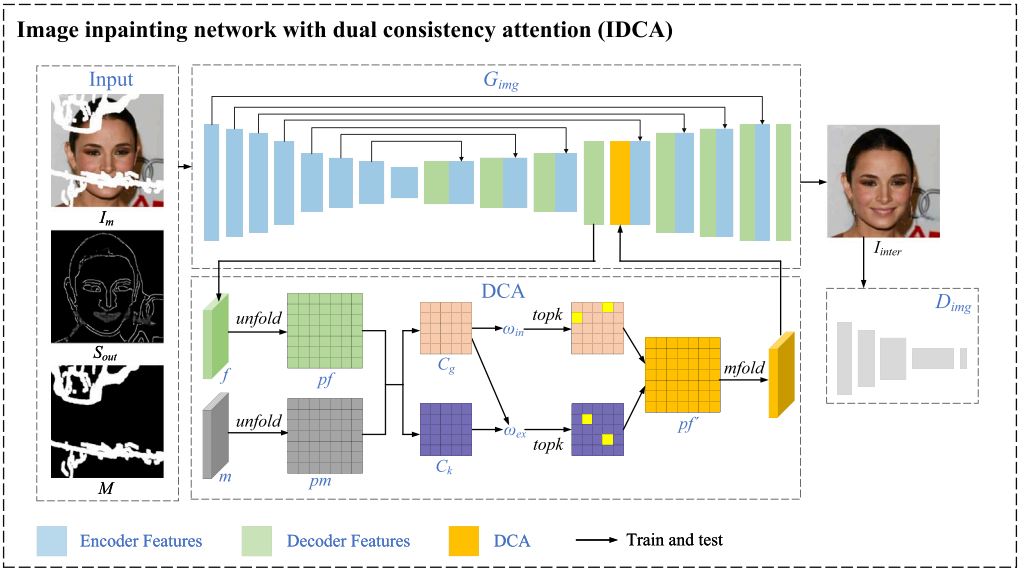

Fig. 3. The specific architecture of IDCA.

known contents (C_k). The internal similarity matrix ω_{in} (between C_g and C_g) and the external similarity matrix ω_{ex} (between C_g and C_k) are computed by normalized inner product. The calculation processes are described as Eqs. (6) and (7).

$$\omega_{in}(i,j) = \langle \frac{C_g(i)}{\|C_g(i)\|}, \frac{C_g(j)}{\|C_g(j)\|} \rangle \tag{6}$$

$$\omega_{ex}(i,j) = \langle \frac{C_g(i)}{\|C_g(i)\|}, \frac{C_k(j)}{\|C_k(j)\|} \rangle \tag{7}$$

where $C_g(i)$ and $C_k(i)$ represent the ith patch of C_g and C_k, respectively. $\omega_{in}(i,j)$ indicates the similarity scores between $C_g(i)$ and $C_g(j)$. Analogously, $\omega_{ex}(i,j)$ is the similarity scores between $C_g(i)$ and $C_k(j)$. In a word, $\omega_{in}(i)$ and $\omega_{ex}(i)$ represent the internal similarity matrix and the external matrix for each patch of C_g ($C_g(i)$), respectively. In $C_g(i)$, the top two patches with the highest similarity in the generated contents and the known contents are separately extracted according to matrixes $\omega_{in}(i)$ and $\omega_{ex}(i)$. Meanwhile, the similarity scores and the positions of these patches are recorded to update $C_g(i)$. The process can be mathematically described as Eqs. (8) and (9).

$$weight_1(i), index_1(i) = topk(\omega_{in}(i), 2) \tag{8}$$

$$weight_2(i), index_2(i) = topk(\omega_{ex}(i), 2) \tag{9}$$

where $topk(\omega, 2)$ denotes the process of finding two patches with the highest similarity scores ($weight$) and their corresponding positions ($index$) in the matrix ω. Thus, the update process can be represented as Eq. (10).

$$C_g'(i) = weight_1(i) \odot pf[index_1(i)] + weight_2(i) \odot pf[index_2(i)] \tag{10}$$

These processes are repeated on each patch of generated contents to obtain the processed patches pf'. The processed patches pf' would be transformed into an entire image by a modified fold operation (named as $mflod$). It is designed based on the original $fold(\cdot)$ function. The original $fold(\cdot)$ function provided in the python library can transform these patches into an image. But it does not work in this paper. This is because that the size of these patches is 3×3 and these values contained in the patches will be computed repeatedly during the folding process. This means that the value of each pixel in the generated images is superimposed by the value of multiple patches. To solve this issue, a modified fold process is proposed. First, a matrix (all-1 matrix) with the same size as the processed patches pf' is built, represented as $flag$. Then, an original $fold(\cdot)$ function is conducted over the $flag$ matrix and pf', obtaining img_{flag} and $img_{pf'}$, as shown in Eqs. (11) and (12).

$$img_{flag} = fold(flag) \tag{11}$$

$$img_{pf'} = fold(pf') \tag{12}$$

In this calculation process, the role of img_{flag} is to calculate how many times the pixels at each position have been superimposed.

Thus, the final output images can be calculated as Eq. (13).

$$img = img_{pf'} / img_{flag} \tag{13}$$

img denotes the final output of DCA. In a word, the proposed attention module does not only consider the consistency within the generated contents but also the consistency between the generated contents and the known contents. This can provide more relevant patches as references to optimize the generated content in the missing regions, obtaining high consistency and coherent semantic inpainting results.

The loss function of this network is constructed by reconstruction loss, adversarial loss, perceptual loss and style loss. The reconstruction loss and adversarial loss are similar to those in the SSC, and they are defined as Eqs. (14) and (15).

$$L_r = \left\| I_{inter} - I_{gt} \right\|_2 \tag{14}$$

$$L_{adv} = \min_{G_{img}} \max_{D_{img}} \mathrm{E}_{I_{gt}} \left[\log D_{img}(I_{gt}) \right] + \mathrm{E}_{I_m} \times \left[\log \left(1 - D_{img}\left(G_{img}(I_m, S_{out}, M) \right) \right) \right] \tag{15}$$

The perceptual loss measures the distance between the two images by using pre-trained convolution neural network, in which the pre-trained network could be VGG, ResNet, etc. The definition of perceptual loss is described as Eq. (16).

$$L_{perc} = \mathrm{E}\left[\sum_i \left\| \phi_i(I_{gt}) - \phi_i(I_{inter}) \right\|_1 \right] \tag{16}$$

where ϕ_i is the activation map in the ith layer of selected pre-trained network. VGG network is selected to compute the perceptual loss. ϕ_i corresponds to activation maps from layers relu1_1, relu2_1, relu3_1, relu4_1 and relu5_1 of the VGG-19 network pre-trained on the ImageNet dataset [34] in this paper. The perceptual loss penalizes the inpainting results and makes them closer to the ground truth images

(a) (b)

Fig. 4. (a) obvious color difference; (b) undesirable details.

perceptually. These activation maps are also used to calculate style loss. Its equation is described as Eq. (17).

$$L_{style} = \mathrm{E}\left[\sum_i \left\| G_i\left(I_{gt}\right) - G_i\left(I_{inter}\right)\right\|_1\right] \tag{17}$$

where $G_i(\cdot) = \phi_i(\cdot)\phi_i(\cdot)^T$ is a Gram matrix [35].

To summarize, the total loss used in this network can be denoted as Eq. (18).

$$L_{IDCA} = \lambda_r L_r + \lambda_{adv} L_{adv} + \lambda_{perc} L_{perc} + \lambda_{style} L_{style} \tag{18}$$

where $\lambda_r = 1$, $\lambda_{adv} = \lambda_{perc} = 0.1$ and $\lambda_{style} = 250$.

4.3. Image local refinement network

After the treatment of structure-aware multi-view image inpainting, relatively plausible inpainting results are obtained. The SSC extracts and repairs structural views where strong constraints are designed to guarantee the inpainting accuracy. Based on the guidance of the repaired structure views and DCA, IDCA can generate image results with high consistency and coherent semantics. These networks both have large receptive fields and can capture more comprehensive image information. However, some missing types (e.g. the local structures or local textures) and the color difference near the holes' boundaries could be better refined when catching information from surrounding regions.

Thus, it is not optimal to pursue a very large receptive field in all cases, especially when repairing local structures and details. This would lead to apparent holes boundaries, undesirable details, as shown in Fig. 4. From the Fig. 4(a), it can be found that the generated contents are related to the known contents, but their color is different from the known contents, resulting in obvious holes' boundaries. Fig. 4(b) indicates that the generated details are unsatisfactory and the repaired image can be easily identified as the generated image. This does not meet the requirements of image inpainting.

The image refinement part is used to address this issue, which is implemented by an image local refinement network (ILR). ILR is implemented based on the combination of a generator G_l and a discriminator D_l. Its specific architecture is shown in Fig. 5. The figure illustrates that ILR is constructed by a shallow network with a small receptive field. The small receptive fields can make the network focus on the surrounding information. This network consists of original convolution and residual blocks. Its input is the combination of the results of IDCA (namely I_{inter}) and the mask images M. The coarse intermediate results I_{inter} would be further refined in a sliding window manner. In this process, the generated images are appropriately refined by using the surrounding local information. The results of ILR are expressed as I_l. To maintain known contents unchanged, the final image inpainting results can be represented as $I_{out} = I_l \odot M + I_m \odot (1 - M)$.

The goal of ILR and IDCA is to make the inpainting results closer to the ground truth. With this in mind, the training objective of ILR (L_{ILR}) uses the same loss functions and hyper-parameters with L_{IDCA}, while only replaces I_l with I_{inter} at the corresponding locations.

5. Experimental results and evaluation

5.1. Datasets and experimental settings

All experiments are conducted on two datasets: Paris StreetView dataset and CelebA dataset.

Paris StreetView [36]: A dataset contains 15,000 street images. Following the original setting, 14,900 images are used for training and the rest for testing.

CelebA [37]: A large-scale face dataset containing 202,599 celebrity images. According to the original evaluation partitions, there are 162,770 images for training and 19,962 images for testing.

To simulate damage in actual situations, the irregular mask dataset proposed by Liu et al. [15] is used in this paper. The used mask dataset contains 12,000 images. These images can be classified into six groups based on the size of the damage relative to the entire image in 10% increments, namely 0%–10%, 10%–20%, 20%–30%, 30%–40%, 40%–50%, and 50%–60%. Each group consists of 2000 images, where 1750 images are used for training and 250 images are used for testing in this paper. All images are resized to 256×256 in the course of experiments. The proposed generative image inpainting method SM-DCA is implemented using PyTorch framework, and the model is optimized by Adam optimizer [38] with $\beta_1 = 0$ and $\beta_2 = 0.9$. The computer with NVIDIA RTX A5000 and 32 GB RAM is used to complete experiments. The batch size is set to 4 during the training. Our model is trained in an "end-to-end" manner. For the learning rate schedule, it is initially set as 1×10^{-4} for the first 500,000 iterations and lowered to 5×10^{-5} in the next 200,000 iterations. In this paper, we compare the proposed model with other five methods: EdgeConnect (EC) [18], Pluralistic Image Completion (PIC) [24], PEN-Net (PEN) [25], BAT [39] and SWT [40].

5.2. Quantitative comparison

Three common evaluation metrics for image inpainting, peak signal-to-noise ratio (PSNR) [41], structural similarity index (SSIM) [42], and mean absolute error (MAE) [43] are used in this section. Since the three metrics can only measure the closeness between the generated images

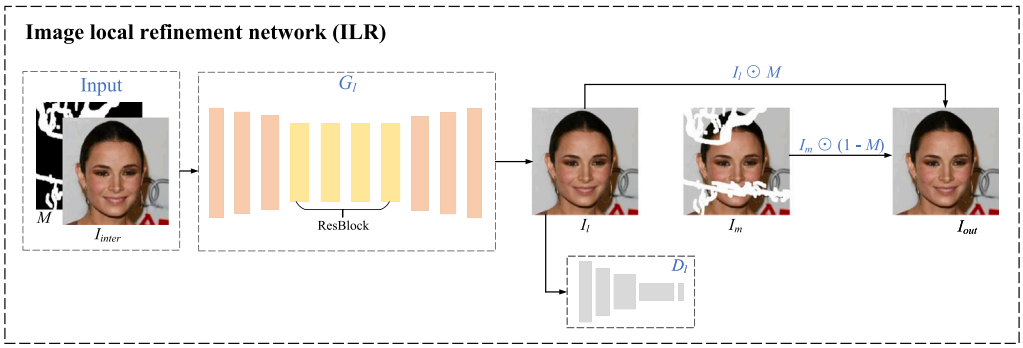

Fig. 5. The specific architecture of ILR.

Table 1
Quantitative results of SM-DCA with five comparison methods on Paris StreetView. ↑Higher is better; ↓Lower is better. **Bold** means the 1st best; Underline means the 2nd best; *Italics* means the 3rd.

Metrics		Methods					
		EC[2019]	PIC[2019]	PEN[2019]	BAT[2021]	SWT[2023]	SM-DCA
PSNR↑	(0.0,0.1]	*35.5251*	34.1424	34.7325	35.6750	32.7786	**36.5037**
	(0.1,0.2]	*30.3147*	29.4310	29.9230	30.4027	30.4004	**31.4235**
	(0.2,0.3]	*27.3137*	26.6520	26.6176	26.9143	27.9044	**28.1372**
	(0.3,0.4]	*25.2185*	24.9409	24.5112	24.7393	25.9435	**25.9756**
	(0.4,0.5]	*23.6249*	23.4725	22.8427	22.9754	**24.3925**	24.3770
	(0.5,0.6]	*21.6041*	21.3820	21.0296	20.7812	21.8530	**22.0194**
SSIM↑	(0.0,0.1]	0.9870	0.9807	0.9852	*0.9869*	0.9788	**0.9889**
	(0.1,0.2]	*0.9610*	0.9473	0.9572	0.9605	0.9611	**0.9669**
	(0.2,0.3]	*0.9197*	0.8977	0.9066	0.9124	0.9273	**0.9276**
	(0.3,0.4]	*0.8804*	0.8544	0.8546	0.8701	**0.8926**	0.8921
	(0.4,0.5]	*0.8382*	0.8033	0.7967	0.8196	**0.8566**	0.8541
	(0.5,0.6]	*0.7342*	0.6931	0.6793	0.7067	0.7400	**0.7457**
MAE↓	(0.0,0.1]	0.0048	0.0076	0.0069	*0.0063*	0.0199	**0.0042**
	(0.1,0.2]	0.0119	0.0156	0.0139	*0.0124*	0.0240	**0.0102**
	(0.2,0.3]	0.0218	0.0258	0.0244	*0.0223*	0.0306	**0.0190**
	(0.3,0.4]	0.0316	0.0353	0.0354	*0.0323*	0.0378	**0.0276**
	(0.4,0.5]	0.0427	0.0461	0.0482	*0.0442*	0.0459	**0.0372**
	(0.5,0.6]	0.0610	0.0658	0.0671	0.0651	*0.0646*	**0.0557**
FID↓	(0.0,0.1]	*6.6252*	12.5879	9.4754	6.2945	11.3564	**5.7338**
	(0.1,0.2]	*16.9133*	26.0090	26.0543	15.5520	17.8386	**15.4908**
	(0.2,0.3]	31.8700	54.4605	54.9452	**29.1339**	*30.9207*	29.9606
	(0.3,0.4]	46.6864	64.8904	81.9085	*43.2813*	**40.5433**	42.8112
	(0.4,0.5]	57.6867	88.0636	103.7667	**50.8977**	54.8266	*55.3092*
	(0.5,0.6]	*81.2369*	107.0700	135.8606	**73.9387**	81.4971	77.4234

Table 2
Quantitative results of SM-DCA with five comparison methods on CelebA. ↑Higher is better; ↓Lower is better. **Bold** means the 1st best; Underline means the 2nd best; *Italics* means the 3rd.

Metrics		Methods					
		EC[2019]	PIC[2019]	PEN[2019]	BAT[2021]	SWT[2023]	SM-DCA
PSNR↑	(0.0,0.1]	*36.8594*	36.5548	37.2780	36.6306	33.5022	**39.6152**
	(0.1,0.2]	31.3624	31.7333	*31.4326*	30.4403	31.1355	**33.5696**
	(0.2,0.3]	28.1237	28.5517	27.7666	26.8187	*28.5383*	**29.9515**
	(0.3,0.4]	25.9201	*26.3095*	25.1152	24.3446	26.3608	**27.4521**
	(0.4,0.5]	24.2846	*24.4716*	23.1761	22.5796	24.6611	**25.6228**
	(0.5,0.6]	21.4500	*21.4254*	20.3910	20.0389	21.4210	**22.5526**
SSIM↑	(0.0,0.1]	0.9905	0.9903	0.9925	*0.9907*	0.9870	**0.9946**
	(0.1,0.2]	0.9757	*0.9775*	0.9782	0.9736	0.9764	**0.9848**
	(0.2,0.3]	0.9546	*0.9555*	0.9520	0.9451	0.9570	**0.9681**
	(0.3,0.4]	*0.9322*	0.9299	0.9180	0.9123	0.9330	**0.9481**
	(0.4,0.5]	0.9068	0.8973	0.8770	0.8753	*0.9055*	**0.9255**
	(0.5,0.6]	0.8370	0.8143	0.7815	0.7977	*0.8186*	**0.8623**
MAE↓	(0.0,0.1]	0.0040	0.0059	*0.0052*	0.0055	0.0180	**0.0027**
	(0.1,0.2]	0.0099	0.0107	*0.0105*	0.0112	0.0211	**0.0070**
	(0.2,0.3]	0.0179	0.0179	*0.0188*	0.0197	0.0263	**0.0133**
	(0.3,0.4]	*0.0263*	0.0258	0.0290	0.0297	0.0327	**0.0203**
	(0.4,0.5]	*0.0353*	0.0350	0.0404	0.0404	0.0397	**0.0279**
	(0.5,0.6]	0.0543	*0.0547*	0.0632	0.0616	0.0596	**0.0453**
FID↓	(0.0,0.1]	*0.2226*	0.3376	0.3649	0.1582	0.5253	**0.1342**
	(0.1,0.2]	*0.6185*	1.0468	1.6628	0.4676	0.8870	**0.4249**
	(0.2,0.3]	*1.1972*	2.5900	5.1332	**1.0026**	1.6724	1.0443
	(0.3,0.4]	*1.9337*	4.9032	12.0150	**1.6574**	2.8706	1.8866
	(0.4,0.5]	2.9713	8.2385	23.3162	**2.3902**	4.6125	*3.0319*
	(0.5,0.6]	5.5230	14.1850	38.9182	**2.9444**	9.5609	*5.7165*

and the ground truth at the pixel level, Fréchet inception distance (FID) [44] is also used to evaluate the inpainting performance at the visual perception. The quantitative results of SM-DCA and the five comparison methods on Paris StreetView and CelebA are presented separately in Tables 1 and 2. From these tables, it can be found that the proposed method SM-DCA achieves the best performance at most cases in terms of PSNR, SSIM, and MAE. BAT shows better performance than SM-DCA at FID in some mask ratios. This is mainly because BAT models the output dependency to align the future predictions with previously predicted tokens and improves the consistency of the reconstructed structures [39]. But the other three metrics of BAT are smaller than those of SM-DCA. In summary, SM-DCA has the best inpainting effectiveness.

5.3. Qualitative comparison

To comprehensively evaluate the performance of SM-DCA, it is necessary to evaluate the generated images not only in numerical experiments but also in terms of visual effects. In this way, the visualization results generated by SM-DCA and the five comparison methods are presented in this section, as shown in Fig. 6. The first five rows are samples from Paris StreetView, and the remaining are samples belonging to CelebA. From this figure, it is clear that our proposed method, SM-DCA, yields satisfactory results, while other comparison methods produce results with some issues such as color inconsistency, uncorrelated textures, distorted structures, blurred details, etc. The red boxes label the differences between these inpainting results. The differences can be seen more clearly by zooming in on this figure. For example, in the first row, the results of EC, PIC, and PEN present color inconsistency and blurry textures. The output of BAT contains unreasonable details, while that of SWT is marred by color artifacts. In the second row, only our proposed method repairs the window that most closely matches the ground truth. In the third row, it is clear that EC, PIC and PEN suffer from color inconsistencies and unreasonable inpainting contents. The result generated by BAT appears to be coherent, but the repaired window still owns distorted railings. While in the output of SWT, there is a problem with the sequential order of walls and railings in the output of SWT. The window, door frames, and walls highlighted by red boxes in the fourth and fifth rows also reflect the similar conclusion above. This implies that the inpainting results generated by our method SM-DCA are more plausible and reasonable. That is, SM-DCA outperforms other methods on Paris StreetView.

Fig. 6. Qualitative results of SM-DCA and other five comparison methods. The numbers below each image indicate PSNR/SSIM.

As for CelebA dataset, the face repaired by SM-DCA in the sixth row is the closest to the ground truth, such as right eye, eyebrow, and nose in red boxes. In the seventh row, the hair, teeth and lip repaired by EC and SM-DCA appear to be broadly plausible. However, the mouth's right corner repaired by EC lacks coordination with the entire lip structure. Similarly, in the eighth row, only our proposed method SM-DCA achieves accurate inpainting, the repaired eyes and mouth are close to the real image. The inpainting results of all other methods exhibit noticeable issues. The same case also appears in the final two rows. The numbers below each output (A/B) indicate the value of PSNR and SSIM, respectively. From this figure, it is clear that the results produced by SM-DCA have the best values in terms of PSNR and SSIM. The numerical results and the visual effects demonstrate the superiority of SM-DCA.

5.4. Ablation experiments and analysis

To better prove the effectiveness of SM-DCA, a series of ablation experiments is designed and performed. The experiments setting can be described as follows:

Baseline = SSC without strong constraint + IDCA without DCA

Test1 = SSC + IDCA without DCA

Test2 = SSC + IDCA

Test3 = SSC + IDCA + ILR (namely SM-DCA)

In this experimental setting, the combination of the structure inpainting network without strong constraints and the image inpainting network without DCA is considered as the Baseline network. The strong constraints (implemented by multi-level structure loss), the dual consistency attention (DCA) module, and the image local refinement

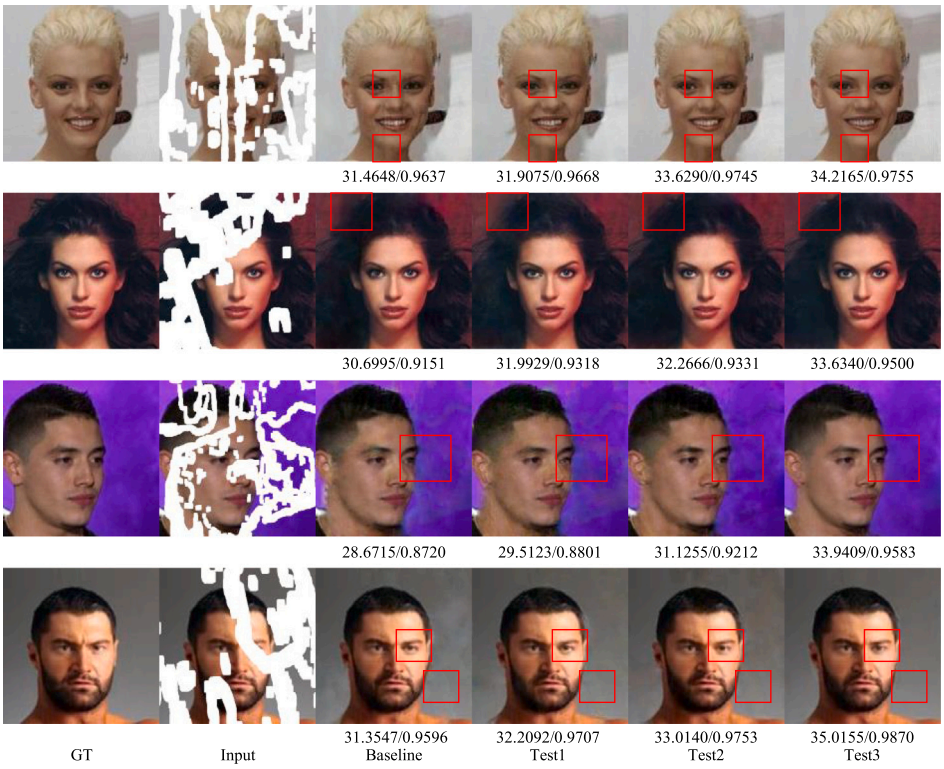

Fig. 7. Qualitative results of ablation experiments. The numbers below each image indicate PSNR/SSIM.

network (ILR) are then successively added to the Baseline. Here, CelebA is selected as the ablation experimental dataset to complete all ablation experiments. The corresponding numerical results are shown in Table 3. From this table, it is obvious that the three modules do improve the inpainting performance in terms of PSNR, SSIM, MAE, and FID. Additionally, the visual results of different experimental settings are shown in Fig. 7. According to the figure, it can be found that the visual effect gradually increases as the three modules are successively added. For the convenience of readers' reading, we have highlighted areas with significant changes using red bounding boxes. The differences can be seen more clearly by zooming in on this figure.

Table 3
Ablation results on CelebA. ↑Higher is better; ↓Lower is better.

Metrics		Methods			
		Baseline	Test1	Test2	Test3
PSNR↑	(0.0,0.1]	36.8188	37.2450	38.3416	**39.6152**
	(0.1,0.2]	31.5900	31.9265	32.6718	**33.5696**
	(0.2,0.3]	28.2485	28.5313	29.1148	**29.9515**
	(0.3,0.4]	25.8999	26.1786	26.6871	**27.4521**
	(0.4,0.5]	24.1515	24.4194	24.8469	**25.6228**
	(0.5,0.6]	21.2808	21.5607	22.0520	**22.5526**
SSIM↑	(0.0,0.1]	0.9916	0.9923	0.9937	**0.9946**
	(0.1,0.2]	0.9782	0.9796	0.9826	**0.9848**
	(0.2,0.3]	0.9565	0.9585	0.9634	**0.9681**
	(0.3,0.4]	0.9317	0.9344	0.9412	**0.9481**
	(0.4,0.5]	0.9035	0.9065	0.9150	**0.9255**
	(0.5,0.6]	0.8346	0.8379	0.8531	**0.8623**
MAE↓	(0.0,0.1]	0.0039	0.0036	0.0032	**0.0027**
	(0.1,0.2]	0.0092	0.0087	0.0079	**0.0070**
	(0.2,0.3]	0.0167	0.0161	0.0150	**0.0133**
	(0.3,0.4]	0.0250	0.0242	0.0230	**0.0203**
	(0.4,0.5]	0.0341	0.0332	0.0323	**0.0279**
	(0.5,0.6]	0.0536	0.0526	0.0500	**0.0453**
FID↓	(0.0,0.1]	0.2407	0.2117	0.1638	**0.1342**
	(0.1,0.2]	0.7296	0.6592	0.4749	**0.4249**
	(0.2,0.3]	1.7315	1.6046	1.1509	**1.0443**
	(0.3,0.4]	3.0810	2.9516	2.1270	**1.8866**
	(0.4,0.5]	4.8527	4.7022	3.4630	**3.0319**
	(0.5,0.6]	7.8214	7.6799	5.8109	**5.7165**

6. Conclusion

A structure-aware multi-view image inpainting method using dual consistency attention method (SM-DCA) is proposed to further improve image inpainting in this paper. This method consists of two parts: structure-aware multi-view image inpainting and image refinement. The former part is implemented by the structure inpainting network with strong constraints (SSC) and the image inpainting network with a dual consistency attention module (IDCA). And the latter part is implemented by the image local refinement network (ILR). In this method, SSC extracts the structure views as additional views to guide the subsequent image inpainting, and makes the repaired structure views close to the real views based on the strong constraints. Guided by the repaired structure views, IDCA completes image inpainting and DCA is designed to consider both internal and external consistency. Then, ILR can refine the inpainting results and generate finer details, eliminating boundary artifacts and further improving the visual quality of the repaired images. Extensive experimental results on Paris StreetView and CelebA demonstrate that SM-DCA can achieve the best inpainting performance.

In this work, we construct a structure view as an additional view to assist image inpainting, where strong constraints are designed to ensure the accuracy of the repaired structure views. In addition, DCA and ILR are proposed to further improve the consistency of the repaired images, generate finer local details, and eliminate artifacts. This work can yield more plausible and reasonable results. The results imply that it is beneficial to design additional views and consider image consistency for the image inpainting task. In the future, we would like to perform joint learning based on image inpainting and other related tasks, such as image super-resolution. It can leverage the complementarities between different tasks to further improve the image inpainting performance, model generalization, and robustness.

CRediT authorship contribution statement

Hongyue Xiang: Conceptualization, Methodology, Software, Validation, Investigation, Writing – original draft. **Weidong Min:** Writing – review & editing, Supervision, Project administration, Funding acquisition. **Qing Han:** Writing – review & editing, Supervision, Funding acquisition. **Cheng Zha:** Investigation, Methodology, Validation. **Qian Liu:** Investigation, Validation, Visualization. **Meng Zhu:** Software, Investigation, Visualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

All used datasets are publicly available.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62076117 and Grant 62166026, and in part by Jiangxi Key Laboratory of Smart City under Grant 20192BCD40002.

References

[1] C. Guillemot, O. Le Meur, Image inpainting, IEEE Signal Process. Mag. 31 (1) (2014) 127–144, http://dx.doi.org/10.1109/MSP.2013.2273004.

[2] X. Zhang, D. Zhai, T. Li, Y. Zhou, Y. Lin, Image inpainting based on deep learning: A review, Inf. Fusion 90 (2023) 74–94, http://dx.doi.org/10.1016/j.inffus.2022.08.033.

[3] J. Shi, N. Xu, H. Zheng, A. Smith, J. Luo, C. Xu, SpaceEdit: Learning a unified editing space for open-domain image color editing, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022), in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE; CVF; IEEE Comp Soc, 2022, pp. 19698–19707, http://dx.doi.org/10.1109/CVPR52688.2022.01911, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, JUN 18-24, 2022.

[4] C. Saharia, J. Ho, W. Chan, T. Salimans, D.J. Fleet, M. Norouzi, Image super-resolution via iterative refinement, IEEE Trans. Pattern Anal. Mach. Intell. 45 (4) (2023) 4713–4726, http://dx.doi.org/10.1109/TPAMI.2022.3204461.

[5] H. Yue, J. Liu, J. Yang, X. Sun, T.Q. Nguyen, F. Wu, Ienet: Internal and external patch matching ConvNet for web image guided denoising, IEEE Trans. Circuits Syst. Video Technol. 30 (11) (2020) 3928–3942, http://dx.doi.org/10.1109/TCSVT.2019.2930305.

[6] A. Criminisi, P. Pérez, K. Toyama, Region filling and object removal by exemplar-based image inpainting, IEEE Trans. Image Process. 13 (9) (2004) 1200–1212, http://dx.doi.org/10.1109/TIP.2004.833105.

[7] C. Barnes, E. Shechtman, A. Finkelstein, D.B. Goldman, PatchMatch: A randomized correspondence algorithm for structural image editing, ACM Trans. Graph. 28 (3) (2009) http://dx.doi.org/10.1145/1531326.1531330, ACM SIGGRAPH Conference 2009, New Orleans, LA, 2009.

[8] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, Image inpainting, in: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, 2000, pp. 417–424.

[9] T. Chan, Nontexture inpainting by curvature-driven diffusions, J. Vis. Commun. Image Represent. 12 (4) (2001) 436–449, http://dx.doi.org/10.1006/jvci.2001.0487.

[10] W. Min, M. Fan, X. Guo, Q. Han, A new approach to track multiple vehicles with the combination of robust detection and two classifiers, IEEE Trans. Intell. Transp. Syst. 19 (1) (2018) 174–186, http://dx.doi.org/10.1109/TITS.2017.2756989.

[11] Q. Wang, W. Min, D. He, S. Zou, T. Huang, Y. Zhang, R. Liu, Discriminative fine-grained network for vehicle re-identification using two-stage re-ranking, Sci. China-Inf. Sci. 63 (11) (2020) http://dx.doi.org/10.1007/s11432-019-2811-8.

[12] H. Zhao, W. Min, J. Xu, Q. Han, W. Li, Q. Wang, Z. Yang, L. Zhou, SPACE: Finding key-speaker in complex multi-person scenes, IEEE Trans. Emerg. Top. Comput. 10 (3) (2022) 1645–1656, http://dx.doi.org/10.1109/TETC.2021.3115625.

[13] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, A.A. Efros, Context encoders: Feature learning by inpainting, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE Comp Soc; Comp Vis Fdn, 2016, pp. 2536–2544, http://dx.doi.org/10.1109/CVPR.2016.278, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, JUN 27-30, 2016.

[14] S. Iizuka, E. Simo-Serra, H. Ishikawa, Globally and locally consistent image completion, ACM Trans. Graph. 36 (4) (2017) http://dx.doi.org/10.1145/3072959.3073659.

[15] G. Liu, F.A. Reda, K.J. Shih, T.-C. Wang, A. Tao, B. Catanzaro, Image inpainting for irregular holes using partial convolutions, in: V. Ferrari, M. Hebert, C. Sminchisescu, Y. Weiss (Eds.), Computer Vision - ECCV 2018, PT XI, in: Lecture Notes in Computer Science, vol. 11215, 2018, pp. 89–105, http://dx.doi.org/10.1007/978-3-030-01252-6_6, 15th European Conference on Computer Vision (ECCV), Munich, GERMANY, SEP 08-14, 2018.

[16] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T. Huang, Free-form image inpainting with gated convolution, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV 2019), in: IEEE International Conference on Computer Vision, IEEE; IEEE Comp Soc; CVF, 2019, pp. 4470–4479, http://dx.doi.org/10.1109/ICCV.2019.00457, IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, SOUTH KOREA, OCT 27-NOV 02, 2019.

[17] S. Li, Q. Liu, Multi-filters guided low-rank tensor coding for image inpainting, in: 2017 2ND International Conference on Image, Vision and Computing (ICIVC 2017), IEEE; Sichuan Prov Comp Sci; Singapore Inst Elect; Chengdu Univ Informat Technol; Chinese Acad Sci Co Ltd, Chengdu Informat Technol, 2017, pp. 418–422, 2nd International Conference on Image, Vision and Computing (ICIVC), Chengdu, PEOPLES R CHINA, JUN 02-04, 2017.

[18] K. Nazeri, E. Ng, T. Joseph, F.Z. Qureshi, M. Ebrahimi, EdgeConnect: Structure guided image inpainting using edge prediction, in: 2019 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), in: IEEE International Conference on Computer Vision Workshops, IEEE; IEEE Comp Soc; CVF, 2019, pp. 3265–3274, http://dx.doi.org/10.1109/ICCVW.2019.00408, IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, SOUTH KOREA, OCT 27-NOV 02, 2019.

[19] S. Xu, D. Liu, Z. Xiong, E2I: Generative inpainting from edge to image, IEEE Trans. Circuits Syst. Video Technol. 31 (4) (2021) 1308–1322, http://dx.doi.org/10.1109/TCSVT.2020.3001267.

[20] Z. Wei, W. Min, Q. Wang, Q. Liu, H. Zhao, ECNFP: Edge-constrained network using a feature pyramid for image inpainting, Expert Syst. Appl. 207 (2022) http://dx.doi.org/10.1016/j.eswa.2022.118070.

[21] W. Xiong, J. Yu, Z. Lin, J. Yang, X. Lu, C. Barnes, J. Luo, Foreground-aware image inpainting, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2019), in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE; CVF; IEEE Comp Soc, 2019, pp. 5833–5841, http://dx.doi.org/10.1109/CVPR.2019.00599, 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, JUN 16-20, 2019.

[22] M.A. Hedjazi, Y. Genc, Efficient texture-aware multi-GAN for image inpainting, Knowl.-Based Syst. 217 (2021) http://dx.doi.org/10.1016/j.knosys.2021.106789.

[23] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T.S. Huang, Generative image inpainting with contextual attention, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE; CVF; IEEE Comp Soc, 2018, pp. 5505–5514, http://dx.doi.org/10.1109/CVPR.2018.00577, 31st IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, JUN 18-23, 2018.

[24] C. Zheng, T.-J. Cham, J. Cai, Pluralistic image completion, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2019), in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE; CVF; IEEE Comp Soc, 2019, pp. 1438–1447, http://dx.doi.org/10.1109/CVPR.2019.00153, 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, JUN 16-20, 2019.

[25] Y. Zeng, J. Fu, H. Chao, B. Guo, Learning pyramid-context encoder network for high-quality image inpainting, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2019), in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE; CVF; IEEE Comp Soc, 2019, pp. 1486–1494, http://dx.doi.org/10.1109/CVPR.2019.00158, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, JUN 16-20, 2019.

[26] H. Liu, B. Jiang, Y. Song, W. Huang, C. Yang, Rethinking image inpainting via a mutual encoder-decoder with feature equalizations, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16, Springer, 2020, pp. 725–741.

[27] J. Peng, D. Liu, S. Xu, H. Li, Generating diverse structure for image inpainting with hierarchical VQ-VAE, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 10775–10784.

[28] J. Zhao, X. Xie, X. Xu, S. Sun, Multi-view learning overview: Recent progress and new challenges, Inf. Fusion 38 (2017) 43–54, http://dx.doi.org/10.1016/j.inffus.2017.02.007.

[29] Y. Xie, B. Lin, Y. Qu, C. Li, W. Zhang, L. Ma, Y. Wen, D. Tao, Joint deep multi-view learning for image clustering, IEEE Trans. Knowl. Data Eng. 33 (11) (2021) 3594–3606, http://dx.doi.org/10.1109/TKDE.2020.2973981.

[30] F. Nie, G. Cai, J. Li, X. Li, Auto-weighted multi-view learning for image clustering and semi-supervised classification, IEEE Trans. Image Process. 27 (3) (2018) 1501–1511, http://dx.doi.org/10.1109/TIP.2017.2754939.

[31] L. Fu, P. Lin, A.V. Vasilakos, S. Wang, An overview of recent multi-view clustering, Neurocomputing 402 (2020) 148–161, http://dx.doi.org/10.1016/j.neucom.2020.02.104.

[32] Y. Ren, X. Yu, R. Zhang, T.H. Li, S. Liu, G. Li, StructureFlow: Image inpainting via structure-aware appearance flow, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV 2019), in: IEEE International Conference on Computer Vision, IEEE; IEEE Comp Soc; CVF, 2019, pp. 181–190, http://dx.doi.org/10.1109/ICCV.2019.00027, IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, SOUTH KOREA, OCT 27-NOV 02, 2019.

[33] P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks, in: 30TH IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE; IEEE Comp Soc; CVF, 2017, pp. 5967–5976, http://dx.doi.org/10.1109/CVPR.2017.632, 30th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, JUL 21-26, 2017.

[34] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. Fei-Fei, ImageNet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252, http://dx.doi.org/10.1007/s11263-015-0816-y.

[35] L.A. Gatys, A.S. Ecker, M. Bethge, Image style transfer using convolutional neural networks, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE Comp Soc; Comp Vis Fdn, 2016, pp. 2414–2423, http://dx.doi.org/10.1109/CVPR.2016.265, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, JUN 27-30, 2016.

[36] C. Doersch, S. Singh, A. Gupta, J. Sivic, A.A. Efros, What makes Paris look like Paris? ACM Trans. Graph. 31 (4) (2012) http://dx.doi.org/10.1145/2185520.2185597.

[37] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: 2015 IEEE International Conference on Computer Vision (ICCV), in: IEEE International Conference on Computer Vision, Amazon; Microsoft; Sansatime; Baidu; Intel; Facebook; Adobe; Panasonic; 360; Google; Omron; Blippar; iRobot; Hiscene; nVidia; Mvrec; Viscovery; AiCure, 2015, pp. 3730–3738, http://dx.doi.org/10.1109/ICCV.2015.425, IEEE International Conference on Computer Vision, Santiago, CHILE, DEC 11-18, 2015.

[38] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.

[39] Y. Yu, F. Zhan, R. Wu, J. Pan, K. Cui, S. Lu, F. Ma, X. Xie, C. Miao, Diverse image inpainting with bidirectional and autoregressive transformers, in: Proceedings of the 29th ACM International Conference on Multimedia, 2021, pp. 69–78.

[40] T.-J. Liu, P.-W. Chen, K.-H. Liu, Lightweight image inpainting by stripe window transformer with joint attention to CNN, 2023, arXiv preprint arXiv:2301.00553.

[41] C. Yim, A.C. Bovik, Quality assessment of deblocked images, IEEE Trans. Image Process. 20 (1) (2011) 88–98, http://dx.doi.org/10.1109/TIP.2010.2061859.

[42] Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli, Image quality assessment: From error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612, http://dx.doi.org/10.1109/TIP.2003.819861.

[43] L. Liu, Y. Liu, Load image inpainting: An improved U-net based load missing data recovery method, Appl. Energy 327 (2022) http://dx.doi.org/10.1016/j.apenergy.2022.119988.

[44] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, Gans trained by a two time-scale update rule converge to a local nash equilibrium, in: Advances in Neural Information Processing Systems, Vol. 30, 2017.