IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 26, 2020, accepted January 11, 2021, date of publication January 18, 2021, date of current version January 25, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3051982

Semantic-SCA: Semantic Structure Image Inpainting With the Spatial-Channel Attention

JINGJUN QIU[1], YAN GAO[2], AND MEISHENG SHEN[1]
[1]Software Engineering Institute, East China Normal University, Shanghai 200062, China
[2]School of Computer Science and Technology, East China Normal University, Shanghai 200062, China

Corresponding author: Yan Gao (ygao@cs.ecnu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 62072183.

ABSTRACT Deep learning has brought unprecedented progress to image inpainting. However, the existing methods often generate images with blurry textures and distorted structures because they may either fail to maintain semantic consistency or restore fine-grained textures. In this paper, we propose a two-stage adversarial model to further improve the accuracy of the structure and details of image inpainting. Our model splits the inpainting task into two parts: semantic structure reconstructor and texture generator. In the first stage, we first utilize the semantic structure map based on the unsupervised segmentation to train the semantic structure reconstructor, which completes the missing structures of the inputs and maintains consistency between the missing part and the overall image. In the second stage, we introduce the spatial-channel attention (SCA) module to obtain the fine-grained textures. The SCA module strengthens the capability to obtain information from the long-distance pixel and different channels of the model. Furthermore, we propose a spatial-channel loss to stabilize the network training process and improve visual effects. Finally, we evaluate our model over the publicly available datasets CelebA, Places2, and Paris StreetView. When the inpainting tasks involved in large-area defects or heavy structure, the experimental results show that our method has a higher inpainting quality than the existing state-of-the-art approaches.

INDEX TERMS Artificial neural networks, deep learning, generative model, image generation, image inpainting.

I. INTRODUCTION

Image inpainting, aim to restoring missing regions according to the rest of the image in image processing, has been widely used in image editing, such as removing unwanted objects and editing contents of images. Recently, deep learning has succeeded in the field of image inpainting. However, how to reasonably extract semantic and structural information of images to obtain accurate and detailed image is still a difficult problem and hot issue in image inpainting.

Early researchers focus on the reconstruction by texture synthesis techniques [1]–[5]. These methods utilize nearest-neighbor searching and copy relevant patches to fill in the missing region with image patches from existing regions. However, these methods have poor performance when there is no repetitive texture available in the undamaged area due to ineffective capturing high-level semantics from the image. In contrast, recent inpainting [6]–[10] research utilize deep convolution neural networks to generate the miss region, and some recent studies treat the inpainting task as a conditional generation problem. Although these methods can achieve plausible inpainting results, they lack correct boundary information to generate the contents of holes, thus the results contain noise patterns and incomplete objects, as shown in Fig. 1. Simultaneously, the range of information extracted by a single convolution layer is too small due to the limitation of the size of the convolution kernel; thus this limitation is not conducive to capturing the global structure information from the long-distance pixel.

The creative process of paintings inspires us. and the artist usually first determines the area of the object and then further fills in the details of different areas in the creative process of painting. Thus, to solve these problems of the over-smoothed boundaries and texture artifacts, we propose a novel method to accurately extract semantic structure information of images. Our model splits the inpainting task into two parts: semantic structure reconstructor and texture generator. The semantic structure reconstructor focuses on

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

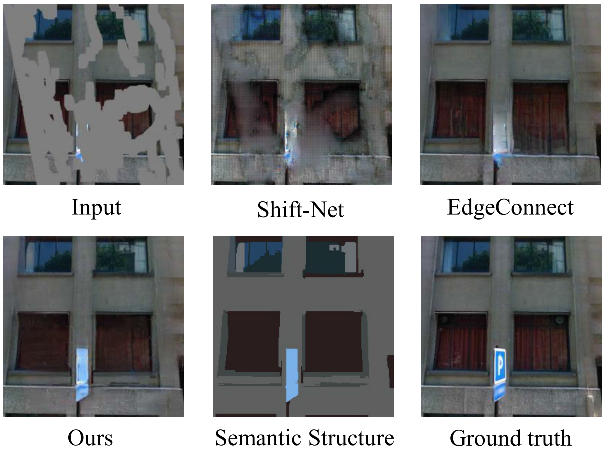

FIGURE 1. **When the missing area is large, the results of these methods [9], [11] of inpainting cannot well maintain the integrity of the semantic structure.**

generating the semantic structure map in the missing areas, and the texture generator uses semantic structure map to generate the final image. Inspired by the previous method [12], we transform semantic segmentation results of this method into a semantic structure extractor to make it satisfy the requirements of image inpainting. The texture difference and spatial distance are taken into account, thus, our method has achieved excellent performance in efficiency and effect. In our method, we can input the original image I_{in} and mask $\mathbf{M}$ into the semantic structure reconstructor to obtain the complete semantic structure map as the image structure. Our method takes the texture difference and spatial distance into consideration in the representation of the image structure. In this way, our model will have more advantages in the reproduction of image details because the semantic structure has more details with enriched types and quantity of labels.

To solve the problem of difficulty in obtaining information from distant pixels result from the limitation of the receptive fields of the convolution kernel and make full use of the feature information of different channels, we introduce the SCA modules to make each pixel is calculated by the element-wise sum in the spatial and channel information in texture generator. In addition, we propose the spatial-channel loss to guide the image generation, which ensures the accuracy of the information of the spatial-channel module.

We conduct experiments on standard datasets CelebA [13], Places2 [14], and Pairs StreetView [15], qualitative and quantitative comparisons show that: when the inpainting tasks involved in large-area defects or heavy structure, our method has a higher repair quality than the existing state-of-the-art approaches. Our paper makes the following contributions:

- We first propose a semantic structure reconstructor based on the unsupervised segmentation to generate the semantic structure map as the global semantic structure information. Meanwhile, we have improved the effect and efficiency of the semantic structure extraction to adapt to image inpainting.
- We introduce the spatial-channel module to strengthen capabilities of obtaining the long-range contextual information and fusing the multi-scale context information of the model in the image inpainting. Meanwhile, to enhance the performance of the spatial-channel module, we introduce the spatial-channel loss to guide the texture generator to generate result.
- A trainable network that combines semantic structure reconstructor and texture generator to fill in missing regions exhibiting fine details.

II. RELATED WORK

A. IMAGE INPAINTING BY TRADITIONAL METHODS

At present, the image inpainting methods can be roughly divided into two categories: traditional methods and deep learning methods. Traditional methods [1]–[5], [16], [17] include diffusion-based and patch-based techniques; these methods mainly use low-level features to repair images. Diffusion-based methods [3], [4] utilize the texture around the missing area to fill the missing region by propagation. Therefore, these can only be used to repair small holes. Patch-based methods [2], [17], [18] can search for similar patches from the remaining areas and copy the patches to the missing areas. Patch-based methods can use remote information to recover the missing region due to not limited by distance; thus, Patch-based methods can recover the image with the large missing area. Patch-based methods usually consume a great number of computing resources in calculating the similarity of patches. Therefore, a fast nearest neighbor searching algorithm is proposed in PatchMatch [2] to reduce computational cost. Furthermore, Hays and Efros [17] propose a data-driven inpainting method, which fills the holes of images by finding the closest image patch in a huge database of photographs gathered from the Web. In summary, the diffusion-based and patch-based methods assume missing patches can be found somewhere in the known regions; thus, they cannot produce novel image contents with meaningful structures.

B. IMAGE INPAINTING BY DEEP GENERATIVE MODELS

About the deep learning methods, after the Generative Adversarial Networks [19], Context Encoders [6] firstly utilizes deep neural networks to generate the missing area. Context Encoders fills the holes by extracting the features from the original image. However, the disadvantage of this method is that the resulting image contains too many visual artifacts. And then, to obtain a more realistic inpainting effect, different people proposed different solutions. Iizuka *et al.* [7] extend the work of Context Encoders and propose local and global discriminators to make the image more realistic. The Shift-Net [9] uses a U-Net architecture with a special shift-connection layer to guide the image generation. Zhang *et al.* [20] regard the semantic image inpainting task as a curriculum learning problem, thus propose a step-by-step repair strategy from outside to inside. This method is able to shrink the corrupted regions in original images progressively. Li *et al.* [21] propose a Recurrent Feature Reasoning module;

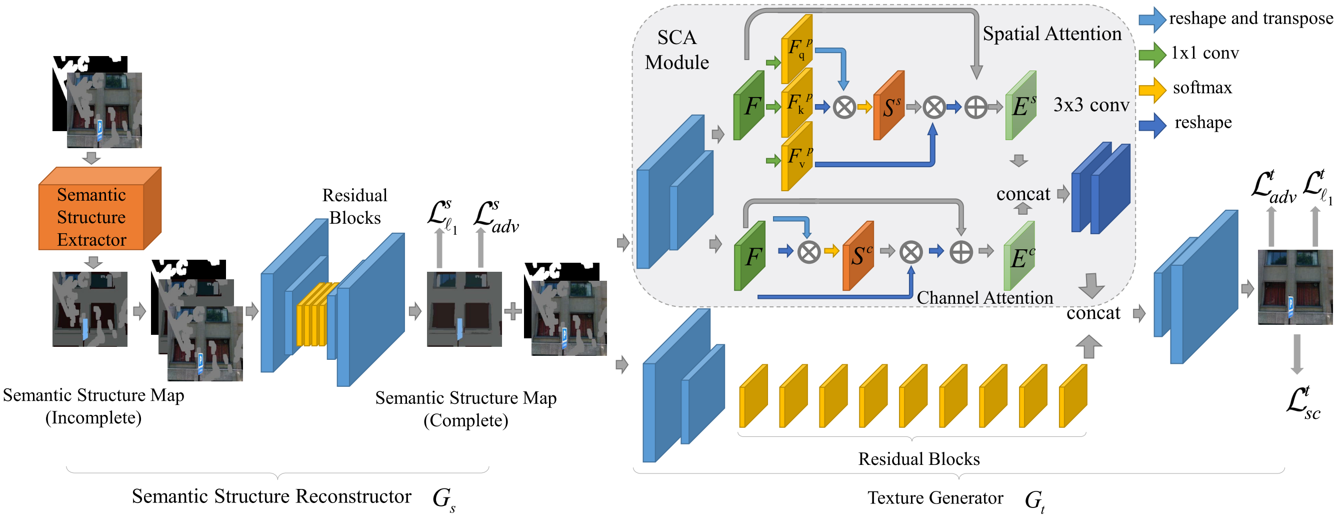

FIGURE 2. **Overview of our inpainting framework(including the framework details of the SCA module). The incomplete image I_{in} and the mask M are fed into the G_s to predict the full semantic segmentation map S_{pred}. The generator G_t takes the mask M and incomplete image I_{in} as input to generate the final result, guided by the semantic structure map S_{pred}.**

the module recurrently infers the hole boundaries of the convolutional feature maps and then uses them as clues for further inference. As the attention mechanism is proposed and applied [22], [23], Liu *et al.* [24] introduce the coherent semantic attention layer to improve the continuity of adjacent pixels. Wang *et al.* [25] introduce a special multistage attention module that considers structure consistency and detail fineness. To generating fine-grained textures, Xie *et al.* [26] and Yu *et al.* [10] introduce the attention mechanism in image inpainting. Xie *et al*. introduce learnable attention maps to update the mask dynamically. Yu *et al*. propose the reason for the image with distorted structures and blurry textures is the ineffectiveness of convolutional neural networks in explicitly borrowing or copying information from distant spatial locations. Therefore they introduce the contextual mechanism to enhance the model of long-term correlations.

To make full use of the mask information, different researchers propose different novel convolution methods [8], [27], [28]. Liu *et al.* [8] propose a partial convolution to distinguish the effective area of the original image. Ma *et al.* [28] propose region-wise convolutions to deal with effective and ineffective regions. To further extract the information from the original image, those methods [29]–[31] introduce the multi-scale mechanism. Reference [30] used two kinds of patches with different sizes to calculate the features separately, and the method combines different features to repair the image. Wang *et al.* [31] propose a network to further extract image features by combining information from different receptive fields.

In network structure, these papers [11], [32]–[34] have introduced the two-stage network. Nazeri *et al.* [11] propose a method named EdgeConnect that recovers the edge of the missing region in the first stage and fills in the missing regions using edge as a priori in the second stage. However, the edge is not an ideal semantic structure because it lost much area information and color information. Moreover, the unclear subordinative relationship between the edge and object will mislead subsequent texture generation, thus edges cannot provide global semantic information. (e.g., It is hard to determine whether those edges belong to a specific object.). In SPG-Net [34], Song *et al*. introduce additional manual labels but this is not available in the practical application, thus, this method can not be used in image inpainting. On the other hand, there are few types of manual labels in the supervised dataset (e.g., The dataset *Cityscapes* [35] used in SPG-Net [34] has only 33 types of labels, and an image may only contain 6-7 kinds of labels), and coarse labels cannot satisfy the demands of image inpainting to generate fine-grained textures. Furthermore, the area with the same semantic labels may have different textures in image inpainting(e.g., continuous but different windows), thus the same label will mislead the process of inpainting in these areas. And then, Liao *et al.* [36] propose a self-evaluation mechanism for image inpainting through segmentation confidence scoring to localize the predicted pixels in the supervised dataset. Recently, some researchers utilize explicit image structure knowledge for inpainting. StructureFlow [37] applies a two-stage model that splits the inpainting task into two stages: structure reconstruction and texture generation. Yang *et al.* [38] introduce a structure embedding scheme which can explicitly provide structure preconditions for image completion.

At the same time, some people think that the result of image inpainting should not be unique; thus, Cai and Wei [39] and Zheng *et al.* [40] proposed those methods that can obtain multiple reasonable results from one original image.

III. OUR APPROACH

The framework for our inpainting network is shown in Fig. 2. Our model consists of two stages: the semantic structure

Algorithm 1 Semantic Structure Extractor

```
Input: $\mathcal{I} = \{i_n \in \mathbb{R}^3\}_{n=1}^{N}$
Output: $\mathcal{J}\{j_n \in \mathbb{Z}\}_{n=1}^{N}$
  $Net.parameter = Init(Xaiver)$
  $\{S_k\}_{k=1}^{K} = PreSeg(\{i_n\}_{n=1}^{N})$
  for $t = 1$ to $T$ do
    $\{x_n\}_{n=1}^{N} = Net(\{i_n\}_{n=1}^{N})$
    for $k = 1$ to K do
      $j_{max} = argmax|j_n|_{n \in S_k}$
      $c'_n = c_{max} \quad for \quad n \in S_k$
    end for
  end for
  $\mathscr{L} = Softmax(x_n, c'_n)$
  $Net.SDG(\mathscr{L})$
```

reconstructor G_s and the texture generator G_t. The semantic structure reconstructor G_s is responsible for generating the complete semantic structure map of the image. The texture generator uses G_s output as a global structure to guide the texture generator G_t to generate fine-grained texture and output the final image.

A. SEMANTIC STRUCTURE RECONSTRUCTOR

The goal of segmentation is to make the image simplified and meaningful, and its results can represent the global semantic structure in image inpainting well. To deal with the resulting image with incomplete objects, we introduce the semantic segmentation map as global semantic structure guidance to generate realistic images.

1) SEMANTIC STRUCTURE EXTRACTOR

Our semantic structure extracting algorithm is as shown in Algorithm 1. Firstly, according to the texture similarity and spatial distance, we pre-classify different areas of the image by the pre-classification. Then, the deep neural network is trained to approximate the result of the pre-classification [41] and merge similar regions.

We compared several different pre-classification algorithms(the experimental detail as shown in Section IV-B4), and finally we employ Felzenszwalb [41] as the pre-classification algorithm.

The autoencoder network structure of semantic structure extractor is shown in Fig. 3. Inspired by SENet [42], we propose a three-layer segmentation network with alternating 3 × 3 convolution kernels and 1 × 1 convolution kernels. To make the picture input to ReLU close to the normal distribution $N(0, 1)$, we put batch normalization before ReLU by referring [43].

2) THE ENCODER-DECODER OF GENERATOR

Let $\mathbf{I}_{gt}$ be the ground truth image, and the $\mathbf{S}_{gt}$ is the semantic structure map (form semantic structure extractor) of $\mathbf{I}_{gt}$. Image mask will be denoted by $\mathbf{M}$, and image mask $\mathbf{M}$ as a pre-condition (1 for the missing region, 0 for background). Therefore, the input of the incomplete image is $\mathbf{I}_{in} = \mathbf{I}_{gt} \odot (1 - \mathbf{M})$, and $\mathbf{S}_{in} = \mathbf{S}_{gt} \odot (1 - \mathbf{M})$ denotes the input of the incomplete semantic structure map. Here, $\odot$ denotes the Hadamard product. The process of our semantic structure reconstructor G_s can be expressed as

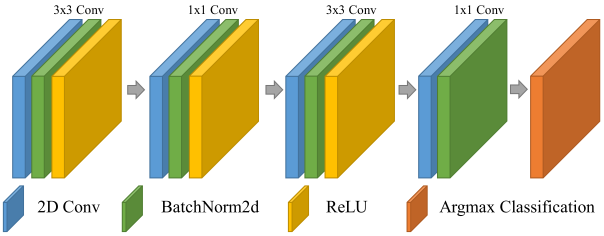

FIGURE 3. Overview of our CNN network framework in the semantic structure extractor, we use alternating 3 × 3 convolution kernels and 1 × 1 convolution kernels in our network.

$$\mathbf{S}_{pred} = G_s(\mathbf{I}_{in}, \mathbf{S}_{in}, \mathbf{M}) \tag{1}$$

where $\mathbf{S}_{pred}$ is the predicted semantic structure map of the $\mathbf{I}_{gt}$, and $\mathbf{S}_{pred}$ is obtained from $\mathbf{I}_{in}$ and $\mathbf{M}$ by G_s.

We use the reconstruction loss $\mathcal{L}^s_{\ell_1}$ of the G_s to measure the ℓ_1 distance between $\mathbf{S}_{pred}$ and $\mathbf{S}_{gt}$.

$$\mathcal{L}^s_{\ell_1} = \left\| \mathbf{S}_{pred} - \mathbf{S}_{gt} \right\|_1 \tag{2}$$

To make the distribution of $\mathbf{S}_{pred}$ close to the distribution of $\mathbf{S}_{gt}$, we adopt the generative adversarial framework [19] in the generator G_s. Thus, the adversarial loss $\mathcal{L}^s_{adv}$ is defined as:

$$\begin{aligned} \mathcal{L}^s_{adv} = & \mathbb{E}[\log(1 - D_s(G_s(\mathbf{I}_{in}, \mathbf{S}_{in}, \mathbf{M}))] \\ & + \mathbb{E}[\log D_s(\mathbf{S}_{gt})] \end{aligned} \tag{3}$$

where D_s is the discriminator of the semantic structure reconstructor. Therefore, the complete loss of the semantic structure reconstructor is:

$$\min_{G_s} \max_{D_s} \mathcal{L}^s(G_s, D_s) = \lambda^s_{\ell_1} \mathcal{L}^s_{\ell_1} + \lambda^s_{adv} \mathcal{L}^s_{adv} \tag{4}$$

where $\lambda^s_{\ell_1}$ and λ^s_{adv} are hyperparameters.

B. TEXTURE GENERATOR

The framework for our texture generator network is shown in Fig. 2. Unlike the traditional methods that mainly rely on copying, the deep learning method extracts features through convolution for image inpainting. However, the receptive field in each layer of convolution is limited by the size of the convolution kernel. It is difficult to establish a strong relationship between pixels that are far apart. Furthermore, after multiple layers of convolution and pooling, it is impossible to reconstruct fine-grained objects from a theoretical level due to the loss of information. To solve the above problems, we introduce the SCA modules to make each pixel is calculated by the element-wise sum in the spatial and channel information in image completion. Inspired by [44], to establish the connection between distant pixels, we extend

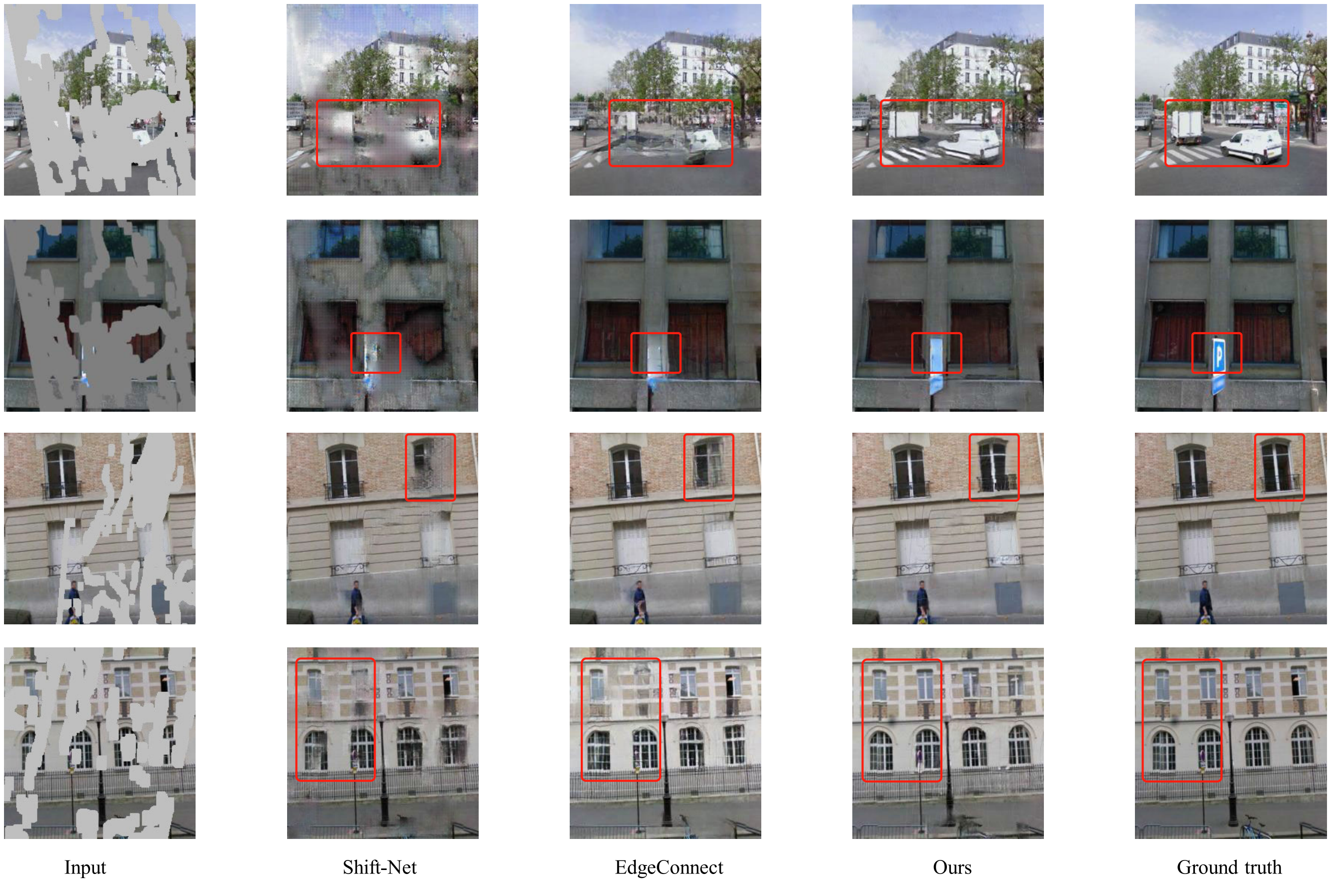

FIGURE 4. Qualitative comparisons on Paris StreetView dataset. Results of Shift-Net [9], EdgeConnect [11] and Ours. By comparison, we can conclude that our method has obvious advantages in maintaining the integrity of objects.

the self-attention mechanism in the task of image inpainting. In order to establish the relationship between channels and obtain the different features of different channels, we extend a self-attention mechanism for channels in image inpainting by referring [42], [45].

The texture generator network G_t employs $\mathbf{S}_{pred}$ as the global semantic structure from G_s, the mask $\mathbf{M}$ and $\mathbf{I}_{in}$ as input to yield realistic result results. The processing of G_t can be defined as:

$$\mathbf{I}_{pred} = G_t(\mathbf{I}_{in}, \mathbf{S}_{pred}, \mathbf{M}) \tag{5}$$

where $\mathbf{I}_{pred}$ denotes the final image output. We use a joint loss to ensure that the generated image is realistic enough. The joint loss consists of reconstruction loss $\mathcal{L}^t_{\ell_1}$, spatial-channel loss $\mathcal{L}^t_{pc}$, and adversarial loss $\mathcal{L}^t_{adv}$. The reconstruction loss $\mathcal{L}^t_{\ell_1}$ of G_t is:

$$\mathcal{L}^t_{\ell_1} = \left\| \mathbf{I}_{pred} - \mathbf{I}_{gt} \right\|_1 \tag{6}$$

and the adversarial loss of G_t is:

$$\begin{aligned}\mathcal{L}^t_{adv} = \mathbb{E}[\log(1 - D_t(G_t(\mathbf{I}_{in}, \mathbf{S}_{pred}, \mathbf{M})))] \\ +\mathbb{E}[\log D_t(\mathbf{I}_{gt})]\end{aligned} \tag{7}$$

In our work, we introduce the spatial-channel loss $\mathcal{L}^t_{sc}$ in texture generator; we will explain this loss in detail in Section III-B2. Finally, taking reconstruction loss, spatial-channel loss, adversarial loss into account, our overall loss of G_t is:

$$\begin{aligned}\min_{G_t}\max_{D_t} \mathcal{L}^t(G_t, D_t) = \lambda^t_{\ell_1}\mathcal{L}^t_{\ell_1} + \lambda^t_{\ell_{sc}}\mathcal{L}^t_{sc} \\ +\lambda^t_{adv}\mathcal{L}^t_{adv}\end{aligned} \tag{8}$$

where $\lambda^t_{\ell_1}$, $\lambda^t_{\ell_{sc}}$, λ^t_{adv} are hyperparameters.

1) SCA MODULE

In the SCA module, we calculate spatial attention and channel attention separately, and integrate these two kinds of attention together by 3 × 3 convolution.

a: SPATIAL ATTENTION CALCULATION

As shown in Fig. 2, we first feed $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$ into a 1 × 1 convolution layer to obtain $\left\{\mathbf{F}^p_q, \mathbf{F}^p_k, \mathbf{F}^p_v\right\} \in \mathbb{R}^{C\times H\times W}$. The purpose of using 1 × 1 convolution is to reduce the number of channels in spatial attention to reduce the computational burden, therefore we use $\mathbf{F}^p_v \cdot S^p$ to restore the number of channels to the value input by this module. Next, we reshape the $\mathbf{F}^p_q, \mathbf{F}^p_k$ to $\mathbb{R}^{C\times N}$, where $N = H \times W$ is the number of the

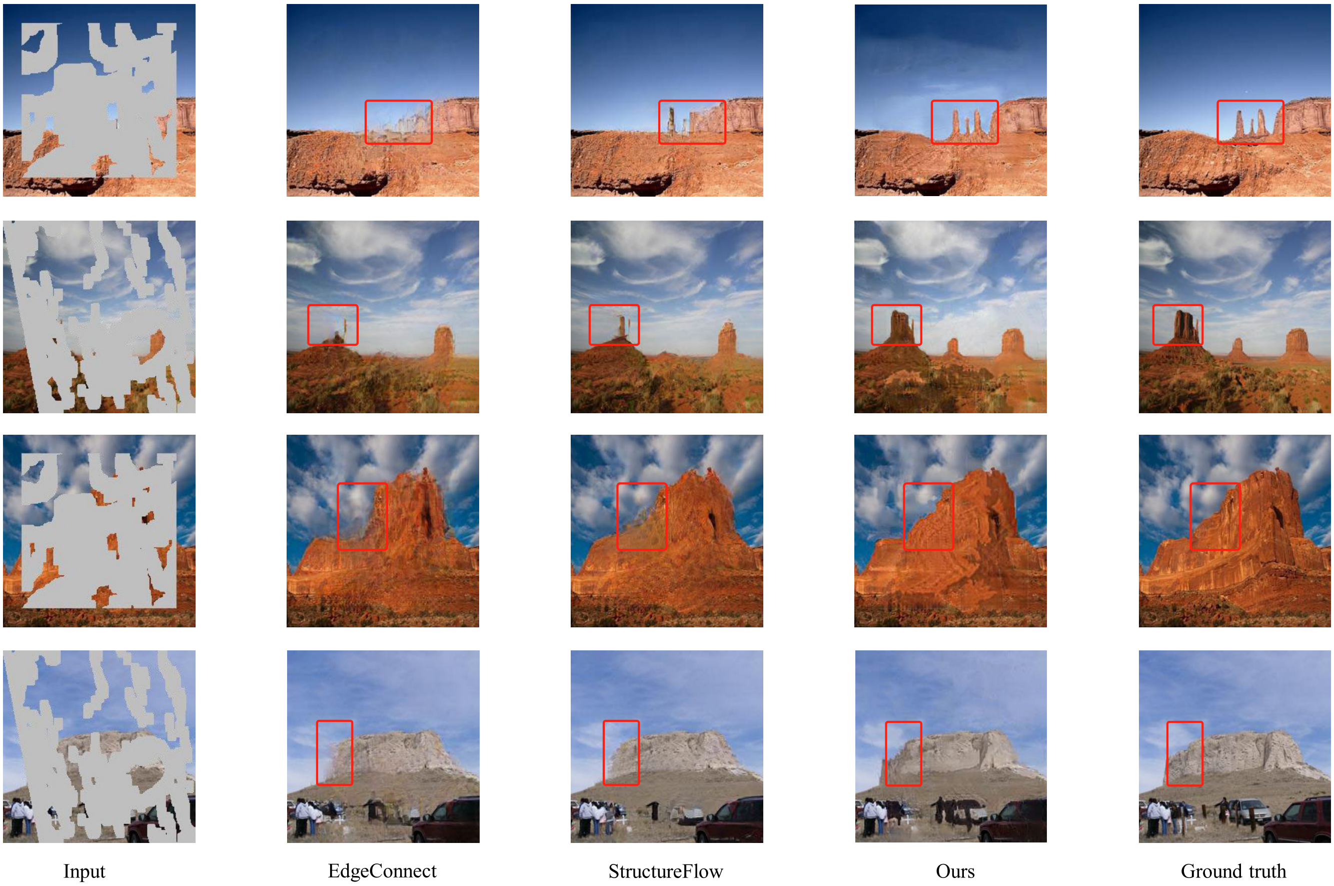

FIGURE 5. Qualitative comparisons on Place2 dataset. Results of EdgeConnect [11], StructureFlow [37] and Ours. By comparison, we can conclude that our method has obvious advantages in restoring the edges of objects in the image inpainting.

pixels. And then we feed $\mathbf{F}_q^p \cdot \mathbf{F}_k^{p\,\mathrm{T}}$ into a softmax layer to get the spatial attention map $S^p \in \mathbb{R}^{N\times N}$.

$$s_{ji}^{p} = \frac{\exp(F_{q\,i}^{p} \times F_{k\,j}^{p})}{\sum_{i=1}^{N} \exp(F_{q\,i}^{p} \cdot F_{k\,j}^{p})} \tag{9}$$

where s_{ji}^{p} represents similarity of the i'th position and j'th position. Meanwhile, after reshaping the $\mathbf{F}_v^p$ to $\mathbb{R}^{C\times N}$, we perform a matrix multiplication between $\mathbf{F}_v^p$ and the transpose of $\mathbf{S}^p$, and reshape the result to $\mathbb{R}^{C\times H\times W}$, we multiply the result by α, and then perform an element-wise sum to get $\mathbf{E}^p \in \mathbb{R}^{C\times H\times W}$:

$$E_j^p = \alpha \sum_{i=1}^{N} (s_{ji}^{p} F_{v\,i}^{p}) + F_j \tag{10}$$

where α is initialized as 0 and gradually learns to assign more weight [23]. Therefore, this means that each pixel position information is weighted by all pixel information. The purpose of the above operation is to use the learned long-distance dependency to act on the original map F to strengthen the global dependency of local features selectively.

b: CHANNEL ATTENTION CALCULATION

Each channel map represents different high-level features of the image, so we introduce the channel attention to explicitly model interdependencies between channels. Because we need to calculate the correlation between channels, we design a different network structure for the channel attention calculation than spatial attention calculation. We remove the 1×1 convolution that reduces the number of channels, and we prove the effectiveness of this operation through experiments (the experiment details are shown in in Section IV-B5).

The framework for channel attention module is shown in Fig. 2. We reshape the $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$ to $\mathbb{R}^{C\times N}$, then after performing a matrix multiplication between $\mathbf{F}$ and the transpose of $\mathbf{F}$, the results $\mathbf{F}^p \cdot \mathbf{F}^{p\,\mathrm{T}}$ through softmax layer to get the channel attention map $\mathbf{S}^c \in \mathbb{R}^{C\times C}$:

$$s_{ji}^{c} = \frac{\exp(F_i \cdot F_j)}{\sum_{i=1}^{N} \exp(F_i \cdot F_j)} \tag{11}$$

where s_{ji}^{c} represents the similarity of the i'th channel and j'th channel. Then, we perform a matrix multiplication of $\mathbf{F}$ and the transpose of $\mathbf{S}^c$. After multiplying $\mathbf{F}$ and $\mathbf{S}^c$ with the parameter β, the result performs an element-wise sum operation with $\mathbf{F}$ to get the $\mathbf{E}^c \in \mathbb{R}^{C\times H\times W}$:

$$E_j^c = \beta \sum_{i=1}^{C} (s_{ji}^{c} F_i) + F_j \tag{12}$$

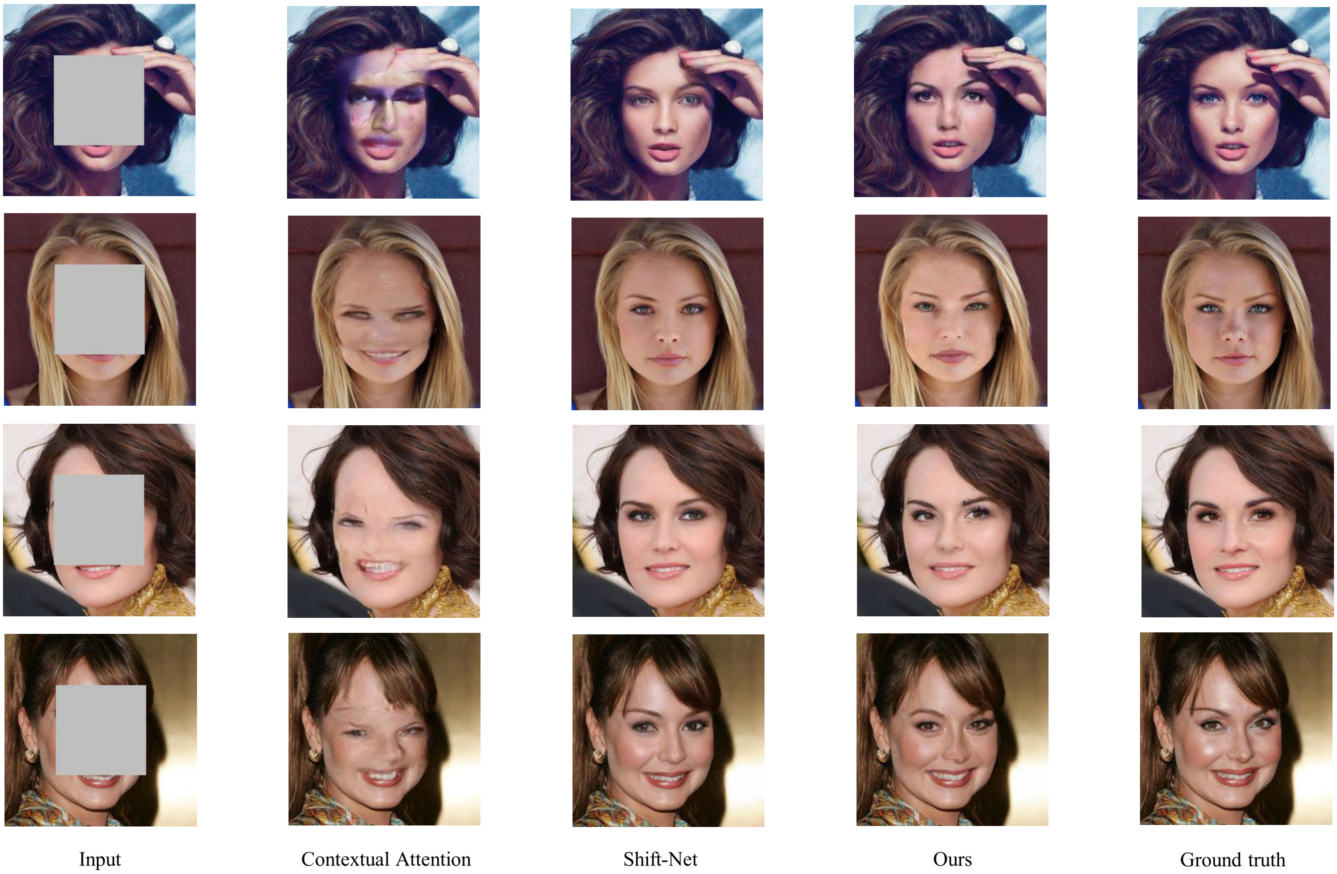

FIGURE 6. Qualitative comparisons in centering masks cases. Results of CA [10], Shift-Net [9] and our method on the Celeba dataset. The experiments show that our method can obtain competitive results in centering inpainting tasks.

where β gradually learns a weight from 0. The final feature map is weighted by all channel information, thus the texture generator can achieve better results in cross-channel information integration capability.

2) SPATIAL-CHANNEL LOSS

To correctly guide the network to obtain the spatial information and channel information, we propose the loss $\mathcal{L}_{sc}$ for the texture generator. We set the spatial feature space and channel feature space as the target for I_{gt} and I_{pred} to compute the $\mathcal{L}_2$ distance.

$$\mathcal{L}_{sc}^{t} = \sum_{y \in M} \|S^{s}(\mathbf{I}_{pred}) - S^{s}(\mathbf{I}_{gt})\|_2 + \left\|S^{c}(\mathbf{I}_{pred}) - S^{c}(\mathbf{I}_{gt})\right\|_2 \tag{13}$$

where S^s and S^c denote the spatial attention map and the channel attention map separately in SCA module(as shown in Fig 2).

IV. EXPERIMENTS

In our work, we evaluate our method on three public datasets: Paris StreetView [15], Celeba [13], and Places2 [14]. Places2 has 1.6 million training images from 365 scene categories, and the scene categories selected from Places2 are *butte*, *canyon*, *field-road*, *field-cultivated*, *field-wild*, *synagogue-outdoor*, *tundra*, *valley*.

A. IMPLEMENTATION DETAILS AND TRAINING

We use the original partition rules of the three datasets to divide the training set, validation set, and test set. The Paris StreetView contains 14,900 training images and 100 test images. We have selected eight categories from Places2. Each category has 5,000 training images, 900 test images, and 100 validation images. The CelebA contains 202599 images. We divide 162770 training images, 19867 validation images, and 19962 test images. And all the dataset image size is 256×256. We obtain irregular masks from the work of PC [8]. These masks are classified based on different hole-to-image area ratios (e.g., 0-10(%), 10-20(%), etc.). The irregular mask dataset includes 55,116 training images and 12,000 test images.

Generators G_s, G_t are trained separately until the losses converge. Several residual blocks [49] are added to further process the features. For semantic structure reconstructor training, given the ground truth image I_{gt}, the incomplete image I_{in}, and the mask $\mathbf{M}$, they were input to the generator G_s to obtain an image S_{pred} of the predicted semantic structure map. Furthermore, the semantic structure reconstructor

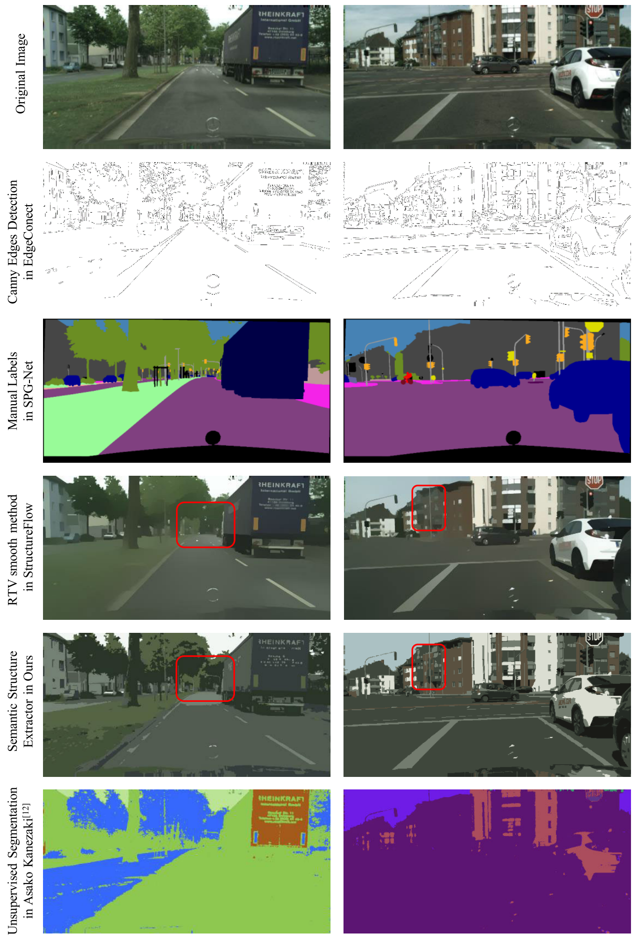
Original Image
Canny Edges Detection in EdgeConnect
Manual Labels in SPG-Net
RTV smooth method in StructureFlow
Semantic Structure Extractor in Ours
Unsupervised Segmentation in Asako Kanezaki[12]

FIGURE 7. Semantic structure Effect Comparisons. Compared with Canny Edges Detection Algorithm [47] in EdgeConnect [11], our method can retain more color and area information. And comparative experimental results between manual labels in SPG-Net [34] and our method show that our method can retain more detailed semantic information. Experimental results show that our method can get a clearer semantic structure map than RTV smooth method [48] in StructureFlow [37]. Relative to the [12], our method can extract more accurate semantic structure information.

applies $\mathcal{L}^{s}_{\ell_1}$ and $\mathcal{L}^{s}_{adv}$ as the guide to update the parameters and weights of the generator G_s to make the predicted semantic structure map S_{pred} of the image as close to semantic structure map S_{gt} of the ground truth as possible. When training texture generator, taking the ground truth image I_{gt}, the incomplete image I_{in}, the semantic structure map S_{pred}, and the mask $\mathbf{M}$ as input, the generator G_t outputs the final result. After that, we finally perform joint training on G_s and G_t.

Like other deep learning methods, we only use the mask $\mathbf{M}$ and the incomplete image I_{in} to generate the semantic structure map S_{pred} completely during the testing time. Then, we use the semantic structure map S_{pred}, the mask $\mathbf{M}$, and the incomplete image I_{in} to complete the final image repair and get the final repair result I_{pred}.

Our model was trained on a single NVIDIA GTX 1080Ti(11GB) with a batch size of 16. We used the Adam algorithm [50] to optimize our model with a learning rate of 1×10^{-4} and $\beta_1 = 0.5, \beta_2 = 0.9$. The training of CelebA [13] model, Places2 [14] model, and Pairs StreetView [15] model took roughly four days, six days, and three days respectively.

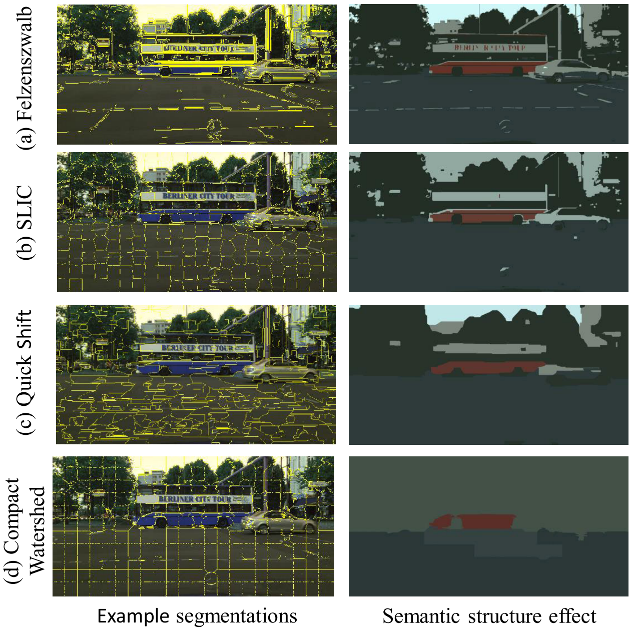

FIGURE 8. The impact of different pre-classification methods on semantic structure map. (a) Felzenszwalb [41] (b) SLIC [51] (c) Quick Shift [52] (d) Compact Watershed [53].

B. COMPARISONS

For better evaluation, we conducted experiments on both settings of centering and irregular masks, and we get the irregular mask from the work of [8]. The inpainting results of the comparison come from the public pre-trained model from these methods [9]–[11].

1) QUALITATIVE COMPARISONS

The irregular inpainting results of the qualitative comparison are shown in Fig. 4 and Fig. 5. We can find that compared with Shift-Net [9], EdgeConnect [11] can get better texture details. However, we also find that these methods cannot maintain the semantic integrity of the object. For example, in the comparison result of the first row and the second row in Fig. 4, our method can better maintain the integrity of the object and achieve better results. In the comparison results of the third row and the fourth row in Fig. 4, our method has less distortion and achieves clearer results (e.g., the repair results of windows). In Fig. 5, compared with EdgeConnect [11] and StructureFlow [37], we can clearly find that our method has the advantages of maintaining object integrity and restoring the edges of the object.

The centering inpainting results of the comparison are shown in Fig. 6. Compared with the CA [10], the Shift-Net [9] has achieved very realistic results, and the results of the Shift-Net are smoother than ours. In the comparison of results, our model can get clearer and sharper inpainting results.

TABLE 1. Comparison results with the random hole between CA [10], PConv [8], EdgeConnect [11], StructureFlow [37] and ours in the Places2 dataset. We use structural similarity index (SSIM), peak signal-to-noise ratio (PSNR), and Fréchet Inception Distance (FID) [46] as the quantitative indicators of the model.

	PSNR			SSIM			FID		
Mask	0-20%	20-40%	40-60%	0-20%	20-40%	40-60%	0-20%	20-40%	40-60%
CA	27.150	20.001	16.911	0.9269	0.7613	0.5718	4.8586	18.4190	37.9432
PConv	31.030	23.673	19.743	0.9070	0.7310	0.5325	-	-	-
EdgeConnect	29.972	23.321	19.641	0.9603	0.8600	0.6916	3.0097	7.2635	19.003
StructureFlow	32.029	25.218	21.090	**0.9738**	0.9026	0.7561	2.9420	7.0354	22.3803
Ours	**32.102**	**29.870**	**27.373**	0.9682	**0.9112**	**0.8071**	**2.9396**	**6.9650**	**18.7960**

TABLE 2. Comparison results with centering hole between CA [10], SH [9], CSA [24] and ours in the CelebA dataset. We use structural similarity index (SSIM) and peak signal-to-noise ratio (PSNR) as the quantitative indicators of the model.

	SSIM	PSNR
Contextual Attention	0.882	23.93
Shift-Net	0.926	26.38
Coherent Semantic Attention	0.931	26.54
Ours	**0.933**	**32.23**

TABLE 3. Comparison results with centering hole between SH [9], EdgeConnect [11], StructureFlow [37] and ours in the the Paris StreetView dataset [15]. We use structural similarity index (SSIM) and peak signal-to-noise ratio (PSNR) as the quantitative indicators of the model.

	SSIM	PSNR
Shift-Net	0.923	27.64
EdgeConnect	0.932	29.96
StructureFlow	0.938	30.57
Ours	**0.945**	**31.96**

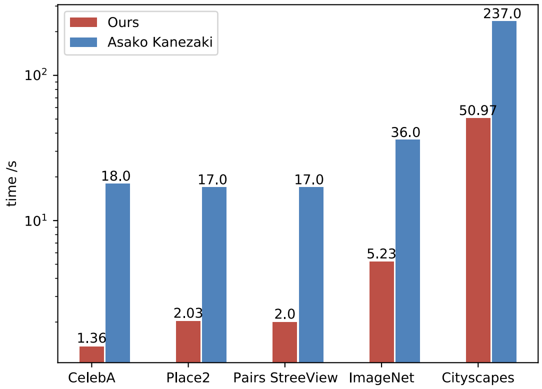

FIGURE 9. The comparison of the time between our approach and [12] for processing an image. The image size is 256 × 256 in on Paris StreetView [15] and Places2 [14], the image size is 500 × 375 in ImageNet [54], and the image size is 2048 × 1024 in Cityscapes [35].

2) QUANTITATIVE COMPARISONS

The results of the quantitative comparison are shown in Table 1, Table 2 and Table 3. For the irregular mask, as shown in Table 1, when the inpainting tasks involved in large-area defects or heavy structure, we can find that our model outperforms all the methods on these measurements with large missing areas (e.g., when the missing area reaches 40-60%). As shown in Table 2 and Table 3, our method can also obtain competitive results compared with existing methods for the centering mask.

3) USER STUDY

Besides, The user study is conducted on Celeba, Paris StreetView, and Places2 for subjective visual quality evaluation. We randomly select 100 images from the test set covering different irregular holes, and the inpainting results are generated by Shift-Net, EdgeConnect, and ours. We invited 30 volunteers to vote for the most visually plausible inpainting result. For each test image, the five inpainting results are randomly arranged and presented to the user along with the input image. The evaluation results are shown in Table 4. Our approach has a better repair effect in repair tasks with obvious boundaries, such as in Paris StreetView and Places2.

4) DIFFERENT STRUCTURE COMPARISONS

As the semantic structure information guiding image generation, the semantic structure map plays an important role in image inpainting. We compared several different structural representations in image inpainting, as shown in Fig. 7. In the generated semantic structure effect, compared with EdgeConnect [11], our method retains more color information and boundary information. Furthermore, our method can better retain the details of semantic structure than SPG-Net [34]. And the structure obtained by our method is more accurate than the method proposed by Kanezaki [12]. Simultaneously, the experimental results of the fourth line in Fig. 7 show that our method can get a clearer structure map than RTV smooth method [48] used in StructureFlow [37]; moreover, as Fig. 9 shows, our method also has a great advantage over the method proposed by AsakoKanezaki *et al.* in the efficiency comparison. Therefore, these prove that our method is more suitable for image inpainting tasks.

At the same time, we also conducted experiments on the impact of different pre-classification methods. As shown in the experimental results in Fig 8, compared with SLIC [51], Quick Shift [52] and Compact watershed [53], we can find that semantic structure extractor with Felzenszwalb [41] can hit more correct edges and get a more precise semantic structure map.

5) THE NETWORK FRAMEWORK IMPROVEMENTS IN THE CHANNEL ATTENTION CALCULATION

In the channel attention calculation, we use a different network structure from the spatial attention calculation. Since we need to calculate the relationship between each channel,

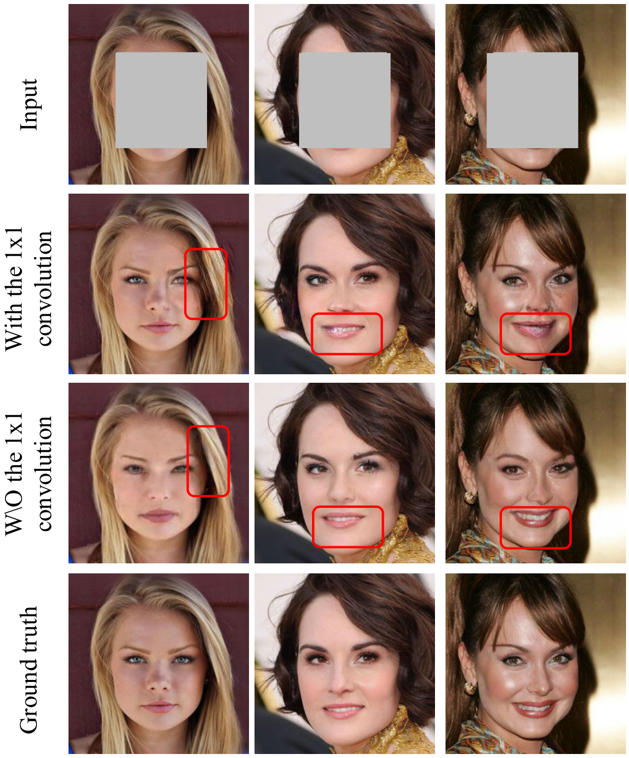

FIGURE 10. The effect of the semantic structure reconstructor. The (a) is the input of this ablation study, and (b),(c) are results of the model without semantic structure reconstructor or with semantic structure reconstructor.

TABLE 4. The evaluation results of the user study. The fooling rate is provided in this table.

	Shift-Net	EdgeConnect	**Ours**
CelebA	15.08%	**25.28%**	23.72%
Pairs	20.06%	32.44%	**34.69%**
Places2	17.72%	26.36%	**27.03%**

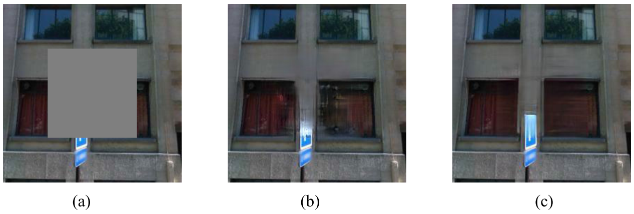

FIGURE 11. The effect of the semantic structure reconstructor. The (a) is the input of this ablation study, and (b),(c) are results of the model without semantic structure reconstructor or with semantic structure reconstructor.

we removed the 1 × 1 convolution used to reduce the number of channels in spatial attention calculation. As shown in the experimental results in Fig 10, we prove that removing the 1 × 1 convolutional can eliminate artifacts generated in image inpainting.

C. ABLATION STUDIES

In this section, we analyze how each component of our model contributes to the final performance from three perspectives: semantic structure, SCA module and the spatial-channel loss.

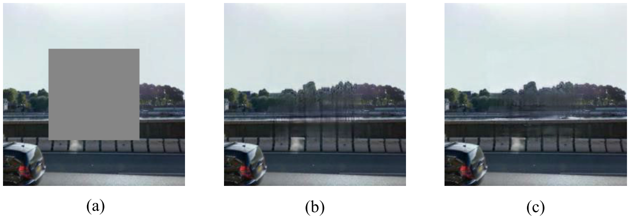

FIGURE 12. The effect of the SCA Module. (b),(c) are results of our model without or with SCA module by input (a).

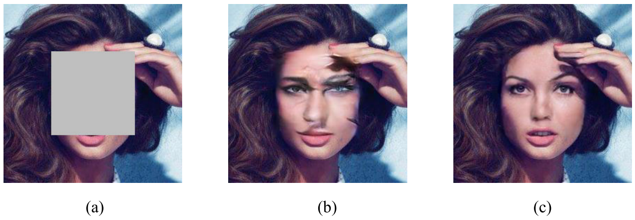

FIGURE 13. The effect of the spatial-channel loss. Given the input (a), the images (b),(c) are the results when using spatial-channel loss and without using the spatial-channel loss, respectively.

TABLE 5. The evaluation results of ablation studies. We use SSIM and PSNR as our criteria.

		SSIM	PSNR
Paris	w/o Semantic Structure Reconstructor	0.902	28.491
	w/o SCA Module	0.879	23.217
	w/o Spatial-Channel Loss	0.865	24.326
	Ours	**0.931**	**31.985**
Celeba	w/o Semantic Structure Reconstructor	0.897	26.889
	w/o SCA Module	0.889	24.311
	w/o Spatial-Channel Loss	0.878	25.628
	Ours	**0.928**	**30.859**

1) SEMANTIC STRUCTURE ABLATION

In our method, we assume that the integrity of the semantic structure is very significant, so in this ablation study, we only use the later stage G_t without the semantic structure map S_{pred} to complete the repair work. The results of this ablation study are shown in Fig 11 and Table 5. And the results present that our semantic structure reconstructor is effective in maintaining the integrity of the semantic structure information in the image inpainting.

2) SCA MODULE ABLATION

To verify the effect of SCA module, we design such an ablation experiment: Remove the spatial and channel module and only keep a single Encoder-Decoder structure in the texture generator G_t. The experimental results are shown in Fig 12 and Table 5. In Fig 12, we found that the results of using SCA module have better performance at the guardrail in generated image than without SCA modules, thus, we prove that the SCA module has certain advantages in generating fine-grained resulting images.

3) SPATIAL-CHANNEL LOSS ABLATION

We conduct further experiments to evaluate the effect of spatial-channel loss. We add and drop out the spatial-channel loss to train the inpainting model. The experimental results are shown in Fig 13 and Table 5. When the model does not have the spatial-channel loss, the generated image presents obvious artifacts. The obvious artifacts may be caused by the lack of guidance of the spatial-channel loss in the image generation process. The spatial-channel loss helps to deal with these issues.

V. CONCLUSION

In this paper, we present a novel deep learning model for image inpainting tasks. We first introduce a new method to restore the semantic structure map based on the unsupervised segmentation and the spatial and channel attention module (SCA module) to complete the image repair. Our model is divided into two stages: semantic structure reconstructor and texture generator. First, the semantic structure reconstructor restores the semantic structure map by the incomplete image and the mask. Experiments demonstrate that the improved semantic structure extractor can well represent the global structure information, and we proved that semantic structure map plays an important role in inpainting tasks by experiments. Then, the texture generator restores the texture detail by SCA module. Furthermore, the spatial-channel loss is introduced in the texture generator to enhance the SCA module learning ability for the ground truth feature distribution and training stability. Finally, we verify that our proposed methods can bring stable performance gain to the final results. Especially when the inpainting tasks involved in large-area defects or heavy structure, the experimental results show that our method has a higher repair quality than the existing state-of-the-art approaches.

REFERENCES

[1] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, Dec. 1999, pp. 1033–1038.

[2] 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, 2009.

[3] M. Bertalmío, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. 27th Annu. Conf. Comput. Graph. Interact. Techn.*, J. R. Brown and K. Akeley, Eds., New Orleans, LA, USA, Jul. 2000, pp. 417–424.

[4] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proc. 28th Annu. Conf. Comput. Graph. Interact. Techn.*, L. Pocock, Ed., Los Angeles, CS, USA, Aug. 2001, pp. 341–346.

[5] M. Wilczkowiak, G. J. Brostow, B. Tordoff, and R. Cipolla, "Hole filling through photomontage," in *Proc. Brit. Mach. Vis. Conf.*, W. F. Clocksin, A. W. Fitzgibbon, and P. H. S. Torr, Eds. Oxford, U.K.: British Machine Vision Association, Sep. 2005.

[6] 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.*, Las Vegas, NV, USA, Jun. 2016, pp. 2536–2544.

[7] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Trans. Graph.*, vol. 36, no. 4, p. 107, 2017.

[8] G. Liu, F. A. Reda, K. J. Shih, T. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *Proc. 15th Eur. Conf. Comput. Vis. (ECCV)*, in Lecture Notes in Computer Science, vol. 11215, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Munich, Germany: Springer, Sep. 2018, pp. 89–105.

[9] Z. Yan, X. Li, M. Li, W. Zuo, and S. Shan, "Shift-net: Image inpainting via deep feature rearrangement," in *Proc. 15th Eur. Conf. Comput. Vis. (ECCV)*, in Lecture Notes in Computer Science, vol. 11218, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Munich, Germany: Springer, Sep. 2018, pp. 3–19.

[10] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 5505–5514.

[11] K. Nazeri, E. Ng, T. Joseph, F. Z. Qureshi, and M. Ebrahimi, "Edgeconnect: Structure guided image inpainting using edge prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops ICCV*, Seoul, South Korea, Oct. 2019, pp. 3265–3274.

[12] A. Kanezaki, "Unsupervised image segmentation by backpropagation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Calgary, AB, Canada, Apr. 2018, pp. 1543–1547.

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

[14] 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.

[15] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros, "What makes paris look like Paris?" *ACM Trans. Graph.*, vol. 31, no. 4, pp. 101:1–101:9, 2012.

[16] 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.

[17] J. Hays and A. A. Efros, "Scene completion using millions of photographs," *Commun. ACM*, vol. 51, no. 10, pp. 87–94, Oct. 2008.

[18] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen, "Image Melding: Combining inconsistent images using patch-based synthesis," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 82:1–82:10, 2012.

[19] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., Montreal, QC, Canada, Dec. 2014, pp. 2672–2680.

[20] H. Zhang, Z. Hu, C. Luo, W. Zuo, and M. Wang, "Semantic image inpainting with progressive generative networks," in *Proc. Multimedia Conf. Multimedia Conf.*, S. Boll, K. M. Lee, J. Luo, W. Zhu, H. Byun, C. W. Chen, R. Lienhart, and T. Mei, Eds., Seoul, South Korea, Oct. 2018, pp. 1939–1947.

[21] J. Li, N. Wang, L. Zhang, B. Du, and D. Tao, "Recurrent feature reasoning for image inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 7757–7765.

[22] L. Mou, Y. Zhao, L. Chen, J. Cheng, Z. Gu, H. Hao, H. Qi, Y. Zheng, A. F. Frangi, and J. Liu, "Cs-net: Channel and spatial attention network for curvilinear structure segmentation," in *Medical Image Computing and Computer Assisted Intervention*, (Lecture Notes in Computer Science), D. Shen, T. Liu, T. M. Peters, L. H. Staib, C. Essert, S. Zhou, P. Yap, and A. R. Khan, Eds., vol. 11764. Shenzhen, China: Springer, Oct. 2019, pp. 721–730.

[23] H. Zhang, I. J. Goodfellow, D. N. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *Proc. 36th Int. Conf. Mach. Learn. (ICML)*, vol. 97, CA, USA, K. Chaudhuri and R. Salakhutdinov, Eds., Long Beach, CA, USA, Jun. 2019, pp. 7354–7363.

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

[25] 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.

[26] C. Xie, S. Liu, C. Li, 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)*, Seoul, South Korea, Oct./Nov. 2019, pp. 8857–8866.

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

[28] 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. (IJCAI)*, S. Kraus, Ed., Macao, China, Aug. 2019, pp. 3123–3129.

[29] 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)*, Honolulu, HI, USA, Jul. 2017, pp. 4076–4084.

[30] 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. (IJCAI)*, S. Kraus, Ed, Macao, China, Aug. 2019, pp. 3748–3754.

[31] 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.*, S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., Montréal, QC, Canada, Dec. 2018, pp. 329–338.

[32] W. Xiong, J. Yu, Z. Lin, J. Yang, X. Lu, C. Barnes, and J. Luo, "Foreground-aware image inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Long Beach, CA, USA, Jun. 2019, pp. 5840–5848.

[33] Y. Song, C. Yang, Z. L. Lin, X. Liu, Q. Huang, H. Li, and C. J. Kuo, "Contextual-based image inpainting: Infer, match, and translate," in *Proc. 15th Eur. Conf. Comput. Vis.*, in Lecture Notes in Computer Science, vol. 11206, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Munich, Germany: Springer, Sep. 2018, pp. 3–18.

[34] Y. Song, C. Yang, Y. Shen, P. Wang, Q. Huang, and C. J. Kuo, "Spg-net: Segmentation prediction and guidance network for image inpainting," in *Proc. Brit. Mach. Vis. Conf.*, Newcastle, U.K., Sep. 2018, p. 97.

[35] 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.*, Dec. 2016, pp. 3213–3223.

[36] L. Liao, J. Xiao, Z. Wang, C. Lin, and S. Satoh, "Guidance and evaluation: Semantic-aware image inpainting for mixed scenes," in *Proc. 16th Eur. Conf. Comput. Vis.*, in Lecture Notes in Computer Science, vol. 12372, A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, Eds. Glasgow, U.K.: Springer, Aug. 2020, pp. 683–700.

[37] 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.*, Seoul, South Korea, Oct./Nov. 2019, pp. 181–190.

[38] J. Yang, Z. Qi, and Y. Shi, "Learning to incorporate structure knowledge for image inpainting," in *Proc. 34th Conf. Artif. Intell. (AAAI)*, New York, NY, USA, Feb. 2020, pp. 12605–12612.

[39] W. Cai and Z. Wei, "PiiGAN: Generative adversarial networks for pluralistic image inpainting," *IEEE Access*, vol. 8, pp. 48451–48463, 2020.

[40] C. Zheng, T. Cham, and J. Cai, "Pluralistic image completion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Long Beach, CA, USA, Jun. 2019, pp. 1438–1447.

[41] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, Sep. 2004.

[42] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 7132–7141.

[43] A. Galloway, A. Golubeva, T. Tanay, M. Moussa, and G. W. Taylor, "Batch normalization is a cause of adversarial vulnerability," *CoRR*, vol. abs/1905.02161, pp. 1–4, Dec. 2019.

[44] X. Wang, R. B. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 7794–7803.

[45] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Long Beach, CA, USA, Jun. 2019, pp. 3146–3154.

[46] 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.*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., Long Beach, CA, USA, Dec. 2017, pp. 6626–6637.

[47] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vols. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[48] L. Xu, Q. Yan, Y. Xia, and J. Jia, "Structure extraction from texture via relative total variation," *ACM Trans. Graph.*, vol. 31, no. 6, p. 139, Nov. 2012.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[50] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds., San Diego, CA, USA, May 2015, pp. 1–15.

[51] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Sâsstrunk, "SLIC superpixels compared to State-of-the-Art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[52] A. Vedaldi and S. Soatto, "Quick shift and kernel methods for mode seeking," in *Proc. 10th Eur. Conf. Comput. Vis.*, in Lecture Notes in Computer Science, vol. 5305, D. A. Forsyth, P. H. S. Torr, and A. Zisserman, Eds. Marseille, France: Springer, Oct. 2008, pp. 705–718.

[53] P. Neubert and P. Protzel, "Compact watershed and preemptive SLIC: On improving trade-offs of superpixel segmentation algorithms," in *Proc. 22nd Int. Conf. Pattern Recognit.*, Stockholm, Sweden, Aug. 2014, pp. 996–1001.

[54] 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.

JINGJUN QIU was born in Shandong, China in 1996. He received the B.Eng. degree from the Shanghai Institute of Technology, in 2018. He is currently pursuing the M.A.Sc. degree with the Software Engineering Institute, East China Normal University, Shanghai, China. His main research interests include image inpainting, generative model, and computer graphics.

YAN GAO received the B.S. degree from the Nanjing University of Aeronautics and Astronautics, in 1993, the M.S. degree from the Wuhan University of Technology, China, in 2002, and the Ph.D. degree from Shanghai Jiao Tong University, in 2006, all in computer science. He is currently an Associate Professor with the School of Computer Science and Technology, East China Normal University, China. His research interests include computer animation and geometric modeling and others.

MEISHENG SHEN was born in Zibo, China, in 1995. He received the B.Eng. degree from the School of Computer Science and Technology, Shandong University of Science and Technology, China, in 2018. He is currently pursuing the M.A.Sc. degree with the Software Engineering Institute, East China Normal University, Shanghai, China. His main research interests include point clouds, machine learning, and computer graphics.