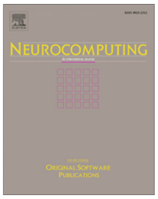
Neurocomputing 536 (2023) 1–12

Contents lists available at ScienceDirect

Neurocomputing

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



Contrastive structure and texture fusion for image inpainting

Long Chen [a,b,c], Changan Yuan [b,c,*], Xiao Qin [c], Wei Sun [b], Xiaofeng Zhu [a,*]

[a] *University of Electronic Science and Technology of China, China*
[b] *Guangxi Academy of Sciences, China*
[c] *Guangxi Key Lab of Human-machine Interaction and Intelligent Decision, Nanning Normal University, China*

ARTICLE INFO

Article history:
Received 21 November 2022
Revised 7 February 2023
Accepted 12 March 2023

Communicated by Zidong Wang

Keywords:
Image inpainting
Contrastive learning
Attention

ABSTRACT

Most recent U-Net based models have shown promising results for the challenging tasks in image inpainting field. However, they often generate content with blurred textures and distorted structures due to the lack of semantic consistency and texture continuity in the missing regions. In this paper, we propose to restore the missing areas at both structural and textural levels. Our method is built upon a U-Net structure, which repairs images by extracting semantic information from high to low resolution and then decoding it back to the original image. Specifically, we utilize the high-level semantic features learned in encoder to guide the inpainting of structure-aware features of its adjacent low-level feature map. Meanwhile, low-level feature maps have clearer texture compared with high-level ones, which can be used as a prior for textural repair of high-level feature maps. subsequently, a module is used to fuse the two repaired feature maps (i.e., structure-aware and texture-aware features) reasonably and obtain a feature map with reasonable semantics. Moreover, in order to learn more representative high-level semantics feature, we design the model as a siamese network for contrastive learning. Experiments on practical data show that our method outperforms other state-of-the-art methods.

1. Introduction

Image inpainting has become one of the research hotspots of computer vision in recent years, which aims at reconstructing the missing or damaged regions of image with reasonable and plausible contents. This technology has been used in many practical applications due to its brilliant performance, such as restoring corrupted photos, photo editing, and object removal (see Fig. 1 for examples). The main challenge in image inpainting lies in using existing image content to generate a reasonable structure and texture for the damaged regions, so that the repaired image appears natural and consistent with the original.

High-quality image inpainting requires an understanding of the contextual information of the damaged image, and filling in the missing area with reasonable structure and texture. Traditional image inpainting methods fill the hole regions using uncorrupted contents of the image by patch-based matching. These methods always fill in the damaged regions pthch-by-patch by searching for well-matched replacement patches in the referable part of the image and copying them to reasonable locations. Barnes et al. [1] propose the Patch-Match method that iteratively searches the best matched patches from boundaries of the hole area to synthesize the contents of the missing parts. Jin and Ye [2] propose a patch-based method based on annihilation property filter and low rank structured matrix. However, these method failure to understand the high-level semantic of the image, resulting in artifacts and blurry contents of the repaired image [3]. As a result, traditional patch-based methods are only applicable to image inpainting with simple contents and single texture.

Recently, learning-based methods have seen tremendous progress and led to state-of-the-art performance in the field of image inpainting due to its ability to effectively extract contextual information. Among these learning-based methods, models similar to encoder-decoder architecture are prevalent. Encoder-decoder likes models first utilize CNN (convolution neural networks) to extract meaningful features from low-level to high-level, and then restore the extracted features to the original resolution. The pioneering learning-based method is [4], which leverages CNN and autoencoder backbone network to both understand the content of the entire image and produce a plausible hypothesis for hole regions. Thereafter, Iizuka S, et al. [5] propose an image inpainting method that automatically restore damaged regions of the image, keeping the local and global contents consistent. Morever, in order to solve the problem that CNN without ability to extract feature

* Corresponding authors at: University of Electronic Science and Technology of China, China (C. Yuan).
E-mail addresses: yca@gxtc.edu.cn (C. Yuan), seanzhuxf@gmail.com (X. Zhu).

https://doi.org/10.1016/j.neucom.2023.03.014
0925-2312/© 2023 Published by Elsevier B.V.

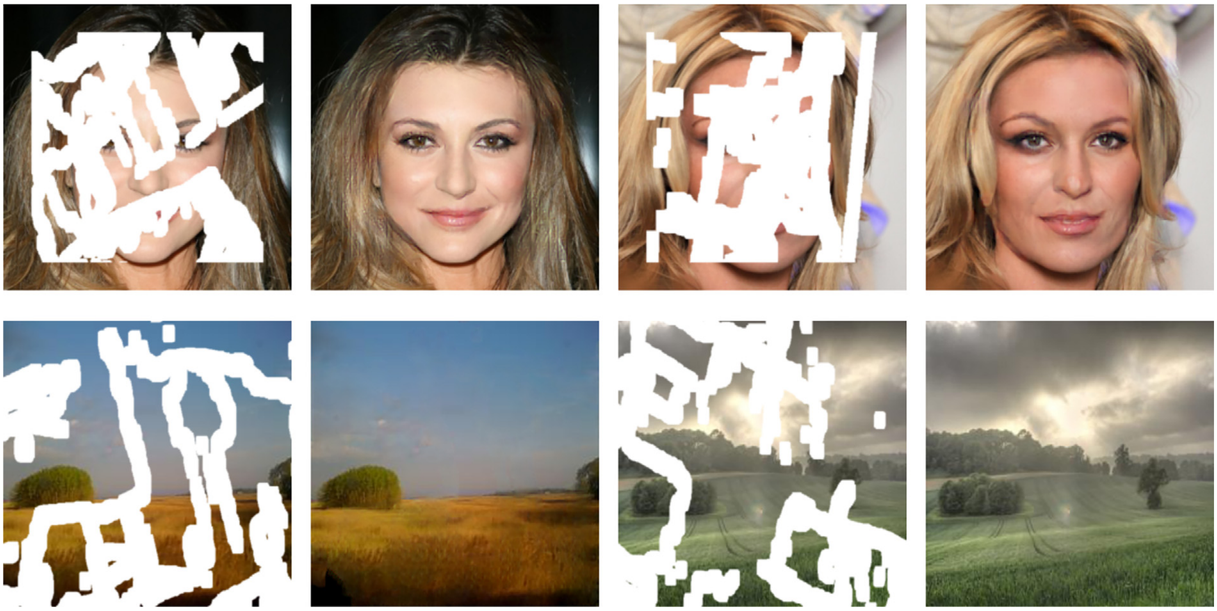


Fig. 1. Selected image inpainting results of our proposed method on CelebA-HQ, Places2 datasets, respectively.

information from hole regions, Yu et al. [6] propose to use an attention module to synthesis contents for hole parts. In anddition, many excellent methods have also been proposed to solve the problem of image inpainting through different ideas. Nazeri K et al. [7] suggest a two-stage adversarial edge connectivity model, which consists of an image generator and an image finishing network. Zeng et al. [8] build a pyramidal-context architecture called PEN-NET for high-quality image inpainting. Liu et al. [9] proposed a layer to the encoder-decoder network called coherent semantic attention (SCA) layer for image inpainting method.

The aforementioned learning-based methods have made great progress in image inpainting and produced promising results compared with traditional methods. However, the damaged parts of the image in real scene is always large, and appear in various unexpected shapes and positions, which greatly increase the difficulty of image inpainting. In other words, without sufficient semantic guidance and the hole area is completely invalid, encoder-decoder models have tremendous challenge to learn powerful features [10,11]. Therefore, these CNN-based methods have to increase the number of networks layers or increase the size of convolution kernel, so as to ensure the strong ability of feature extractor [12]. As a result, this kind of naive way heavily increase the computational cost and can not guarantee that the learned features are accurate [13]. Another shortcoming of CNN-based methods is that convolution networks have ineffectiveness in modeling long-term correlations between distant contents [14,15]. For instance, to allow a pixel being influenced by the content of 64 pixels away, it requires at least 6 layers of 3 × 3 convolutions with dilation factor 2 or equivalent [16]. Furthermore, CNN-based methods fail to fully consider the correlation between structure and texture of the image, resulting in the inconsistent appearances.

To address above issues, we propose a self-supervised siamese network with contrastive learning named Contrastive Structure and Texture Fusion Network (CSTFNet) to conduct image inpainting. Specifically, in order to solve the problem that the hole regions are completely invalid, a two-stage inpainting strategy is introduced to reconstruct structure and texture respectively. these two-stage methods first use an encoder-decoder architecture to generate coarse result in first stage, and design elaborate network for exquisite texture generation in second stage. In fine inpainting phase, CSTFNet simulates the process of image inpainting by human, i.e., predict the structure of the hole regions from high-level semantic features while analyze the texture of the hole area from low-level semantic features. For example, when we repair a damaged image, we will first see the overall contextual information of the image to judge the general structure of the damaged parts, and then fill a reasonable texture according to the know texture around the damaged regions. Consequently, CSTFNet combined with the characteristic of feature extraction of CNN (i.e., the deeper the number of layers represents the higher-level features) repairs the structure and texture respectively. To be specific, CSTFNet actually repair each layer of features in the feature pyramid extracted by CNN. The structural repair of each feature layer is guided by its adjacent higher-level feature while textural repair is affected by its adjacent lower-level feature. Moreover, in order to learn more representative features to guide the repair process of the structure, we propose to use the contrastive learning pattern to train our backbone network. We highlight our contributions as follows.

1. We propose a siamese inpainting network based on contrastive learning for free-form image inpainting, which greatly improves the network's feature learning for damaged images.
2. We propose a novel structure and texture inpainting module that effectively uses high-level features to guide low-level features in repairing the structure and transferring texture details to high-level features.
3. Our method achieves promising inpainting results with rich texture information and reasonable structure on three public datasets against state-o-the-art image inpainting methods.

2. Related Works

2.1. Image Inpainting

Non-learning-based methods. This kind of works mainly fall into two categories: patch-based methods and diffusion-based methods. The patch-based methods mainly copy the content of the background regions to the missing parts by calculating the similarity between patches. The typical patch-based method is proposed by T. Ruºi¢and A. Piºurica [17], which leverage Markov Random Field (MRF) to search the best matching patch. Kawai et al. [18] propose a method based on selecting target objects and restricting search background, which is able to remove the object from an image. Ding et al. [19] proposed a patch-based method for image inpainting using Non-local Texture Matching and Nonlinear Filtering. In order to solve the problem of high computational cost of calculating block similarity, Barnes et al. [1] propose a randomized nearest-neighbor patch matching strategy. These methods mainly are difficult to extract high-level semantic information of images, and thus cannot guarantee the final satisfactory results.

Diffusion-based methods attempt to propagate the known contents to the hole regions under smooth constraints, which is generally used to repair small image damage. The majority of these duffusion-based methods are based on the Partial Differential Equation (PDE) and the variational method [20]. The first to use diffusion-based method for image inpainting is [13] proposed by Bertalmio et al.. They iteratively apply anisotropic diffusion in the process of inpainting the image. Li et al. [21] calculates the diffusion coefficient by considering the distance and direction between the pixel in the hole regions and its neighbor pixels. However, this kind of methods only consider the pixel around the damaged regions, resulting in suffer from blurring artifacts.

Learning-based methods. Learning-based methods have seen a surge in the explosive development of deep learning in recent years. These methods generally extract meaningful semantic information from the damaged images, and fill the damaged parts with reasonable and realistic contents according to the powerful ability of feature learning. The pioneering work to introduce the deep learning into the field of image inpainting is [4], which proposes to use the network structure of combining the encoder-decoder network and generative adversarial network (GAN) [22] for image inpainting. This method mainly extracts and decodes the features of damaged images through CNN, and obtains high fidelity image by leveraging reconstruction loss and adversarial loss to train the model. Iizuka et al. [5] propose to employ two kinds of adversarial loss (i.e., global adversarial and local adversarial loss) to improve the fidelity of the reconstructed images. Whereas, only using CNN to repair large holes will lead to structural distortions and blurring because CNN has no ability to extract the feature of damaged parts. Therefore, in order to fill in the hole regions with reasonable contents, Yu et al. [6] add an attention layer to the vanilla encoder-decoder networks to effectively extract contents from background regions to fill the missing parts. Specifically, method [6] first obtains a rough repaired image through a coarse-inpainting network, then calculates the similarity of contents between hole and background regions. Subsequently, the contents of hole regions are generated by combining the background contents according to the similarity. In addition, there are many excellent learning-based image inpainting methods. Liu et al. [23] propose partial convolutions with automatic mask updating strategy for image inpainting, which enormously reduces the influence of invalid pixels on the results. Liu et al. [24] use CNN to extract the structure and texture features of the image respectively, and then provide supplementary information for the decoder through feature equalization for image generation. Yu et al. [25] propose to use gated convolution in image inpainting to automatically learn the soft mask and the dynamic feature mechanism of each channel at each spatial position from the data.

2.2. Contrastive Learning

Contrastive learning is a way of self supervised learning, which uses positive or negative image pairs to learn representations. It does this by reducing the distance between the representations of positive image pairs while narrowing the similarity of two negative representations [26]. Instance discrimination method [27] generates two positive samples from a any given image through data augmentation, and then inputs the two augmentations into the encoder to obtain two latent representations. Afterwards, the latent representations encoded from the same image are regarded as positive pairs while the representations obtained from different samples are considered as negative pairs. At the same time, contrastive loss is leveraged to draw positive pairs closer and alienate negative pairs. Tian et al. [28] extents contrastive learning to multi-view data and propose a multi-view contrastive learning method named CMC. Specifically, CMC takes the different views of the same sample as positive pairs, and trains model as same as aforementioned instance discrimination method. Chen et al. [29] propose to add a projection layer behind the encoder of contrastive learning, which can learn powerful features that have better performance in downstream tasks. He et al. [30] improve the training of contrastive learning by storing representations from a momentum encoder instead of the trained network [31]. however, creating positive and negative sample pairs requires a great quantity of computing and storage resources. Hence, some studies began to focus on contrastive learning without negative pairs. Grill et al. [32] propose a new method called BYOL that focuses only on the similarity of samples and representations. BYOL transforms the previous contrastive learning from a comparative problem to a prediction problem by introducing a prediction head and stop gradient. In addition, Chen et al. [33] also propose a simple siamese network (SimSiam) to learning high-quality feature representation without using either negative sample pairs or a momentum encoder.

3. Proposed Method

In this section, we introduce our new image inpainting algorithm called CSTFNet (Contrastive Structure and Texture Fusion Network). Fig.2 shows the pipeline of our proposed CSTFNet. Specifically, CSTFNet is a two stage (i.e., coarse-to-fine) network. In the first stage, U-Net like model is leveraged to roughly repair the damaged image and obtain a coarse result. In the second stage, a siamese encoder-decoder network is used for contrastive learning, which repairs two different masked images in a forward process. Moreover, the encoder of siamese network learn feature pyramid from coarse images, and then repair the feature maps of each layer from structural and textural levels respectively. Subsequently, the two kinds of repaired feature maps will be fused through the feature fusion module, and added to each layer of decoders by skip connecting. Finally, high-fidelity images output from refinement network. In the following, we describe the proposed network and loss functions in detail.

3.1. Coarse Inpainting

Fig. 2 shows the network structure of coarse inpainting that similar to U-Net [34]. Our coarse inpainting network (NETc) employs an encoder-decoder architecture with skip connection. Eight downsampling and upsampling operations make up the backbone of the coarse inpainting network (NETc). Specifically, The input of coarse inpainting net (NETc) is a damaged image with its corresponding binary mask. Subsequently, The masked image and mask are sent to encoder of coarse inpainting network (NETc) and then decoded to the coarse repaired image. In the coarse inpainting network (NETc), all convolution operations use convolution kernel with size of 3 and stride of 2. The output of coarse inpainting network (NETc) is often blurred in the hole regions due to the influence of invalid pixel values of miss parts. Subsequently, the coarse image will be sent to refinement net for secondary inpainting.

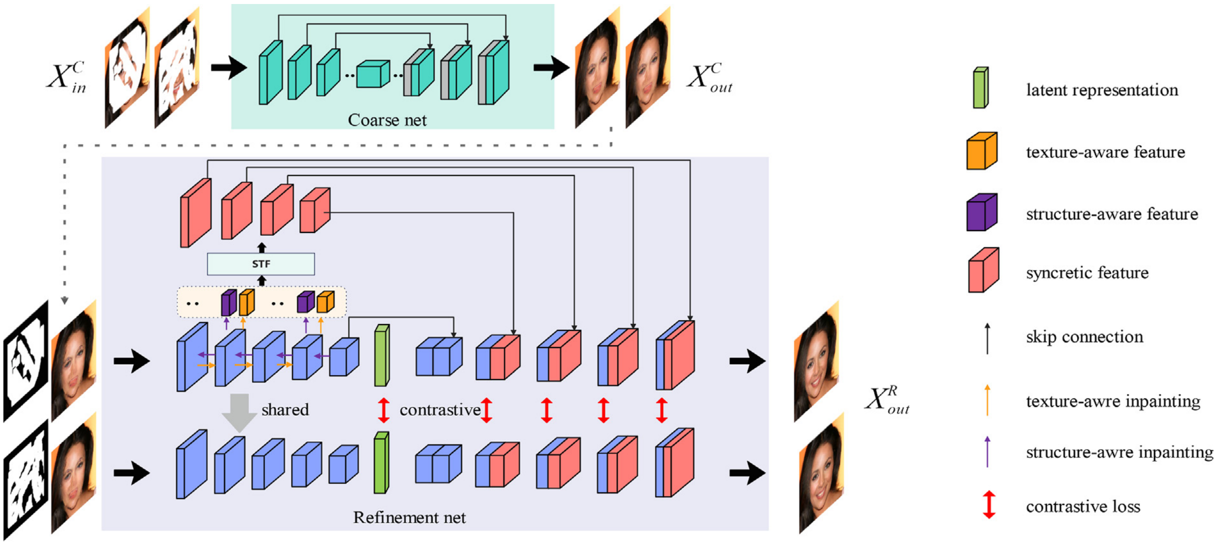


Fig. 2. Overview of our proposed framework.

In the phase of training, we adopt pixel-wise reconstruction loss and adversarial loss to optimize the parameters of coarse inpainting network (NETc). We use L1 loss for the pixel-wise reconstruction.

$$L^C_{hole} = \frac{\left\| \left(X^C_{out} - I_{gt}\right) \odot (\mathbf{1} - M) \right\|_1}{sum(\mathbf{1} - M)} \tag{1}$$

$$L^C_{valid} = \frac{\left\| \left(X^C_{out} - I_{gt}\right) \odot M \right\|_1}{sum(M)} \tag{2}$$

Where L^C_{hole}, L^C_{valid} are reconstruction loss of damaged regions and background regions respectively. I_{gt} is the ground-truth image, M is the binary indicator matrix, and $\odot$ represents element-wise product operation. The overall pixel-wise reconstruction of the whole image is denoted as:

$$L^C_r = L^C_{hole} + \alpha L^C_{valid} \tag{3}$$

Where α is a trade-off parameter.

For adversarial loss, we use the least square loss [35], the corresponding loss functions for the coarse inpainting network and discriminator are define as [36]:

$$I^C_{rec} = I^C_{in} \odot (\mathbf{1} - M) + I^C_{out} \odot M \tag{4}$$

$$L_D = \frac{1}{2} E_{I\sim p_{data}(I)} \left[\left(D(I_{gt}) - 1\right)^2 \right] \frac{1}{2} E_{I_{rec}\sim p_{I_{rec}}(I_{rec})} \left[\left(D\left(I^C_{rec}\right) \right)^2 \right] \tag{5}$$

$$L^C_G = E_{I_{rec}\sim p_{I_{rec}}(I_{rec})} \left[\left(D\left(I^C_{rec}\right) - 1 \right)^2 \right] \tag{6}$$

Where I^C_{rec} is the reconstruction image. The total loss for coarse inpainting network (NETc) is $L_C = L^C_{hole} + \alpha L^C_{valid} + \beta L^C_G$, and we set $\alpha = 6$ and $\beta = 0.1$ in our experiments.

3.2. Refinement Inpainting

The refinement net can be divided into four parts: 1) contrastive learning architecture. In order to extract powerful features and contextual information for structure-aware inpainting, CSTFNet leverages contrastive learning to improve the ability of feature extraction of the encoder. 2) structure-aware and texture-aware inpainting (STI). CSTFNet follow the hypothesis that high-level semantic is able to guide the inpainting of structure for low-level semantic. In addition, texture-aware inpainting simulates the process of structure inpainting. What is different is that this module assumes that low-level semantic can guide the inpainting of high-level semantic in texture features. 3) structure and texture fusion. We propose STF module to fuse the structure-aware and texture-aware information learned from above two modules.

3.2.1. Contrastive Learning

We propose self-supervised contrastive learning network consists of two shared weights encoder and decoder. The core idea of contrastive learning is that two observations of the same concept should produce the same outputs [33]. Base on the distinguishing feature of contrastive learning and the uniqueness of image inpainting task, we design to treat the same image with different masks as a pair of positive samples and use them as input of the siamese network in contrastive learning. The detailed structure of our contrastive learning module is shown in Fig.3. It can be seen from the figure that an image with two different masks (i.e., x_1 and x_2) is fed into the siamese network and then passed through encoder f, projector layer g as well as a predictor MLP head h respectively. Denoting the two output vectors as $p_1 = h(g(f(x_1)))$ and $z_1 = g(f(x_2))$. Hence, the contrastive loss is expresses as minimizing the cosine similarity of the two vectors.

$$D(p_1, z_2) = -\frac{p_1}{\|p_1\|_2} \cdot \frac{z_2}{\|z_2\|_2} \tag{7}$$

Where $\|\cdot\|_2$ is l_2-norm. This is equivalent to the mean squared error of l_2-normalized vectors, up to a scale of 2. Following, we define a symmetrized loss as:

$$l = \frac{1}{2} D(p_1, z_2) + \frac{1}{2} D(p_2, z_1) \tag{8}$$

This is defined for each image, and the total loss is averaged over all images [33]. However, Optimizing the siamese network at the same time will lead to the collapse of the model when two samples are mapped to the same point in the low dimensional space. Therefore, stop-gradient operation is leveraged to guarantee that the model can be trained normally. We implement it by modifying (1) as $D(p_1, stopgrad(z_2))$. This means that z_2 does not backpropagate gradients and update weights during training. Similarly, the form in (2) is implemented as:

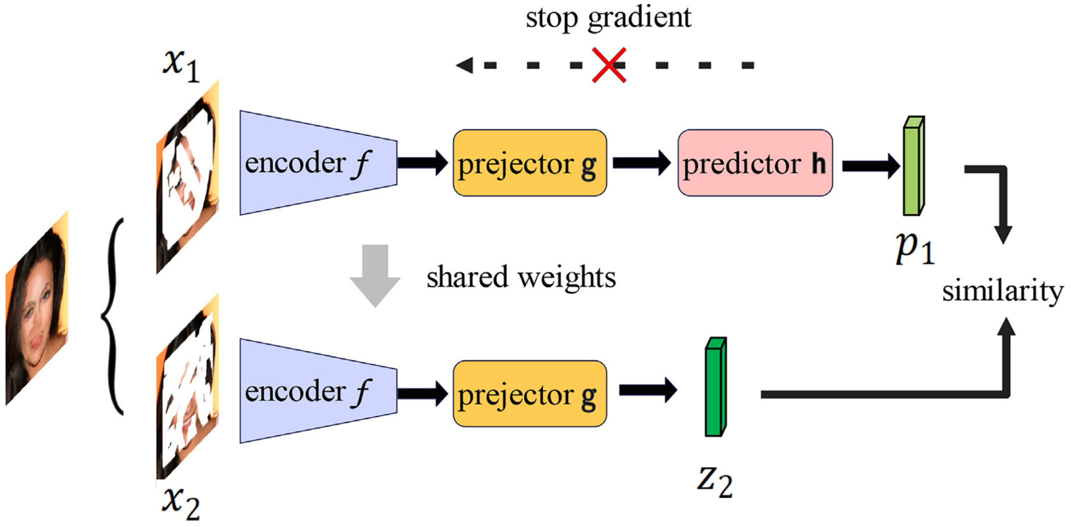


Fig. 3. Contrastive learning module.

$$l = \frac{1}{2}D(p_1, stopgrad(z_2)) + \frac{1}{2}D(p_2, stopgrad(z_1)) \tag{9}$$

Here the encoder on x_1 receives no gradient from z_1 in the first term, but it receives gradients from p_1 in the second term (and vice versa for x_2).

3.2.2. Structure-aware and Texture-aware Features Inpainting

We observe that deep layer extracts high-level semantic information (structural information) while shallow layer extracts detailed information (textural information) in convolution neural networks (CNNs). Base on this characteristic of CNNs, we propose to use two separated branches to repair the hole regions in feature map from structure and texture levels respectively. Specifically, For the feature map of a certain layer in CNNs, we use its adjacent deeper feature map to guide it to repair structure and use its adjacent shallower feature map to guide inpainting of texture. We denote the feature repaired from structural level as structure-aware feature while feature repaired from textural level as texture-aware feature. As shown in Fig.2, structure-aware and texture-aware features are respectively generated at each layer of encoder from the relative direction of CNNs (i.e., texture-aware features are generated from shallow to deep, and structure-aware features are generated from deep to shallow). Under the assumption that pixels with similar semantics should have similar structures and textures, an structure and texture inpainting module is applied at each layer to learn region affinity from high/low-level semantic features, thus the learned region affinity can further guide feature transfer inside/outside missing regions in an adjacent layer with higher/lower resolution.

The specific inpainting process of structure-aware and texture-aware inpainting (STI) is shown in Fig.4, given two adjacent feature maps in feature pyramid. Feature map with large resolution represents low-level feature at shallow layer, while feature maps with low resolution contains higher-level semantic information. In addition, the feature map of each layer contains a part of missing regions (black area) that needs to be repaired because the missing regions do not provide useful information in the convolution process. We define the missing regions in feature map as the result of scaling down the original mask according to the pixel ratio between the feature map and the original image. Moreover, We use attention mechanism to fill the missing regions, that is, the content of the missing regions are formed by combining the content of the background regions according to the attention value.

The attention is usually computed by region affinity between patches outside/inside missing regions. STI module divided each feature map into an equal number of patches, which are respectively from the missing regions and background regions. Subsequently, STI calculates the cosine similarity between patches inside and outside missing regions:

$$a_{mn} = \left\langle \frac{p_m}{\|p_m\|_2}, \frac{p_n}{\|p_n\|_2} \right\rangle \tag{10}$$

Where p_m is the m-th patch extracted from outside missing regions, p_m is the n-th patch extracted from inside the mask. Then softmax is applied on the similarities to obtain the attention score for each patch:

$$a_{n,m} = \frac{\exp(a_{mn})}{\sum_{m=1}^{N} \exp(a_{mn})} \tag{11}$$

After obtaining the attention score from a certain feature map, the holes in its adjacent high/low-level feature map can be filled with context weighted by the attention score. Specifically, the repair of structure-aware features uses the attention score obtained from its adjacent deeper feature map, while the repair of texture-aware features refers to attention from its shallower layer. The repair of feature map can be seen as the process of pixel reconstruction using attention scores:

$$p_n = \sum_{m=1}^{N} a_{n.m} p_m \tag{12}$$

Where p_n is the n-th patch to be filled in missing regions. After calculating all patches, we can finally obtain a filled feature map. In particular, all these operations can be formulated into convolution operations for end-to-end training [6].

3.2.3. Structure-aware and Texture-aware Features Fusion

This module is proposed to further combine the structure-awre and texture-aware features generated by the above STI module. It exchanges messages between the two kinds of information, which fuse structure and texture smoothly. Fig.5 illustrates the proposed STF module. Specifically, we use simple concatenate and convolu-

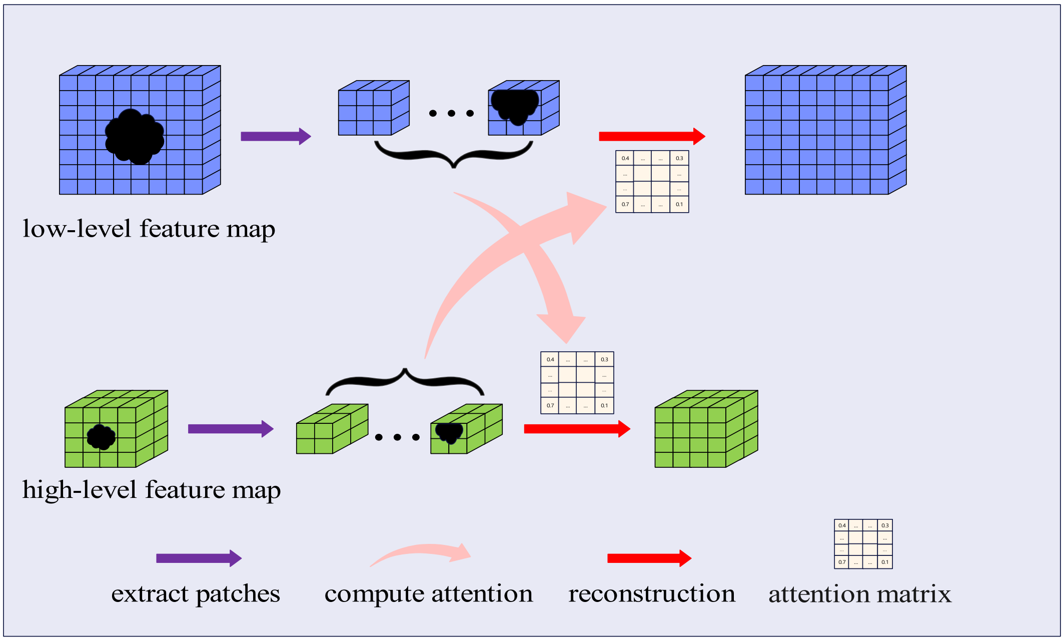


Fig. 4. Structure-aware and texture-aware features inpainting (STI).

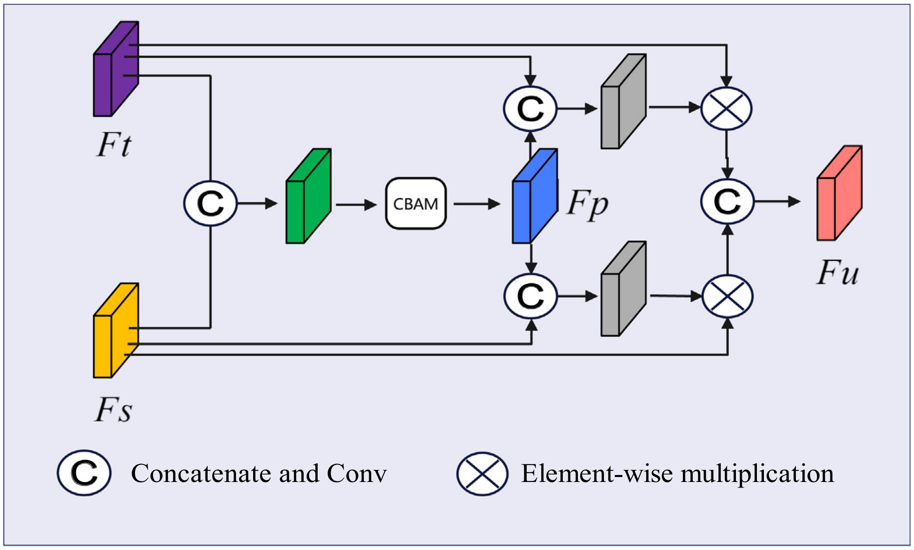


Fig. 5. Structure-aware and texture-aware features fusion (STF).

tion operations to fuse the structure-aware and texture-aware features. Then, through the CBMA [37] module, the preliminarily fused features F_p focus on local information from the spatial and channel levels respectively.

$$F_p = CBAM(Con(C(F_t, F_s))) \tag{13}$$

Where C represents concatenate operation, Con is a convolution layer with the kernel size of 3. After getting the preliminarily fused features F_p, the deep interaction between structure and texture is achieved by concatenate, convolving and multiplying it with structure- and texture- aware features respectively, and finally the integrated feature map is formed.

$$F_u = Con2\left(C\left(F_t \odot Con\left(C\left(F_t, F_p\right)\right), F_s \odot Con\left(C\left(F_s, F_p\right)\right)\right)\right) \tag{14}$$

Where Con is a convolution layer with the kernel size of 3 and Con2 is a convolution layer with the kernel size of 1, $\odot$ denotes element-wise multiplication. The integrated feature F_u obtained from each layer of the encoder will be added to each layer of the corresponding decoder by skip connection.

3.2.4. Loss Function

We train our refinement net with a series of loss functions, including reconstruction loss, perceptual loss, style loss, contrastive loss and adversarial loss so that the finally generated image looks more visually realistic.

Reconstruction Loss. We adopt the l_1 distance as the reconstruction loss to measure the similarity between the final output result X^R_{out} and the ground-truth image, which is the same as Eq. (1) and Eq. (2).

$$L^R_{hole} = \frac{\left\| \left(X^R_{out} - I_{gt}\right) \odot (\mathbf{1} - M) \right\|_1}{sum(\mathbf{1} - M)} \tag{15}$$

$$L_{valid}^{R} = \frac{\left\| \left(X_{out}^{R} - I_{gt} \right) \odot M \right\|_1}{sum(M)} \tag{16}$$

$$L_r^R = L_{hole}^R + \alpha L_{valid}^R \tag{17}$$

Perceptual Loss. We utilize the perceptual loss l_{per} to capture the high-level semantics [38] by computing the l_1 distance between the feature spaces of generated image and groundtruth through ImageNet-pretrained VGG-16 backbone, which can be written as:

$$l_{per} = E\left[\sum_i \left\| \phi_i\left(X_{out}^R\right) - \phi_i(I_{gt}) \right\|_1 \right] \tag{18}$$

Where $\phi_i(\cdot), i = 1, 2, \ldots, 5$ is the feature map of i-th layer of VGG-16, which are ReLu1_1, ReLu2_1, ReLu3_1, ReLu4_1 and ReLu5_1.

Style Loss. In order to reduce the influence of artifacts that resemble checkerboard produced by the transposed convolution layer of the decoder, we introduce style loss To mitigate this effect.

$$l_{style} = E\left[\sum_i \left\| \varphi_i\left(X_{out}^R\right) - \varphi_i(I_{gt}) \right\|_1 \right] \tag{19}$$

where $\varphi_i(\cdot) = \phi_i^T \phi_i$ denotes the Gram matrix constructed from the above-mentioned five activation maps.

Contrastive Loss.

We use contrastive loss to improve the learning ability of the encoder. To be exact, we respectively calculate the contrastive loss between the lantent representations of encoder and the feature maps of each layer of decoder in siamese network. The contrastive loss l_{con}^E of lantent representations is similar to Eq. (9). Moreover, The contrastive loss between the feature maps generated by each layer of decoder is expressed as:

$$l_{con}^D = \sum_i \left\| X_i^1 - X_i^2 \right\|_2 \tag{20}$$

Where X_i^1 and X_i^2 are two feature maps obtained from the decoder of i-th layer in siamese network.

Adversarial Loss. Adversarial loss is mainly used to select the most realistic solution when there are infinite solutions to the ill-posed problem of image inpainting. We choose the same training strategy as that of coarse network for the adversarial loss of refinement network, and the adversarial loss of refinement network is calculated as l_G^R.

To summarize, the objective for refinement network is: $l_R = l_r^R + \lambda_{per} l_{per} + \lambda_{style} l_{style} + \lambda_{con1} l_{con}^E + \lambda_{con2} l_{con}^D + \lambda_G l_G^R$. Therefore, our proposed inpainting network CSTFNet is trained in an "end-to-end" manner, and the final training loss is the summation of losses of two sub-networks and a discriminator, i.e., $l_C + l_R + l_D$.

Algorithm 1: Pseudo code of CSTFNet

```
Input: $X_{in}^C$: input masked image
  $M_{in}$: input mask
Output: $X_{out}^R$: inpainted image
while $i$ smaller than IterNumdo
    $X_{out}^C \leftarrow coarse\_net\left(X_{in}^C, M_{in}\right)$
    $(X_{down1}, .., X_{down6}) \leftarrow Encoding\left(X_{out}^C\right)$
    $(X_{down}^{structure}) \leftarrow Structure\_Inpainting(X_{down})$
    $(X_{down}^{texture}) \leftarrow Texture\_Inpainting(X_{down})$
    $\left(X_{down}^{fusion}\right) \leftarrow STF(X_{down}^{structure}, X_{down}^{texture})$
    $X_{out}^R \leftarrow Decoding\left(skipconnect\left(X_{down}^{fusion}, X_{up}\right)\right)$
    $i \leftarrow i + 1$
end while
return $X_{out}^R$
```

4. Experiment

In this section, we conduct quantitative and qualitative experiments to prove the superiority of our method against state-of-the-

Table 1
Quantitative comparisons of our method with five advanced inpainting methods on CELEBA-HQ dataset. **+** Higher is better. − Lower is better.

	CELEBA-HQ						
	Masks	1–10%	10–20%	20–30%	30–40%	40–50%	50–60%
PSNR**+**	PEN	31.717	26.072	23.748	22.410	21.089	19.456
	RN	35.354	30.514	27.204	24.854	23.608	21.446
	RFR	34.563	28.895	26.303	23.960	22.115	19.553
	CTSDG	36.523	30.570	28.331	26.363	24.774	21.981
	LGNet	37.013	31.120	28.212	26.646	24.691	22.565
	OURs	**37.221**	**31.237**	**28.723**	**26.801**	**24.926**	**22.783**
SSIM**+**	PEN	0.956	0.906	0.857	0.810	0.755	0.691
	RN	0.972	0.958	0.932	0.887	0.839	0.781
	RFR	0.970	0.945	0.905	0.860	0.808	0.730
	CTSDG	0.976	0.957	0.936	0.907	0.871	0.799
	LGNet	0.983	0.960	0.925	0.909	0.868	0.826
	OURs	**0.986**	**0.962**	**0.937**	**0.916**	**0.877**	**0.832**
MSE−	PEN	0.010	0.019	0.031	0.041	0.054	0.073
	RN	0.006	0.012	0.021	0.034	0.042	0.056
	RFR	0.005	0.013	0.022	0.034	0.049	0.075
	CTSDG	0.005	0.012	0.018	0.023	0.031	0.049
	LGNet	**0.004**	0.011	0.019	0.024	0.033	0.045
	OURs	**0.004**	**0.010**	**0.017**	**0.022**	**0.030**	**0.044**
LPIPS−	PEN	0.046	0.106	0.154	0.198	0.245	0.296
	RN	0.051	0.053	0.095	0.124	0.149	0.196
	RFR	0.019	0.056	0.086	0.105	0.139	0.189
	CTSDG	0.019	0.057	0.079	0.091	0.122	0.179
	LGNet	0.018	0.052	0.080	0.069	0.097	0.121
	OURs	**0.017**	**0.049**	**0.074**	**0.064**	**0.090**	**0.120**

art image inpainting methods, and explain our experimental settings, the datasets, competing methods, and implementation details. In addition, we also conduct ablation experiments to evaluate and analyze our proposed model.

4.1. Experimental Settings

Datasets. We evaluate our mothod on three public datasets, which are commonly used for image inpainting tasks.

Places2 dataset [39]: A large-scale scene recognition dataset that contains over 8,000,000 images from over 365 scenes collected from the natural world. We elaborately selected 10000 outdoor landscape images to build our dataset, of which 9000 for training set and the remaining 1000 for testing.

CelebA-HQ dataset [40]: The high-quality version of the CelebA [41] consists of 30,000 face images. We randomly select 27,000 for training and the remaining 3,000 for testing.

Paris StreetView dataset [42]: This dataset consists of 15,000 images collected from street views of Paris. We build training and testing sets according to the original settings.

During training and testing, we used the irregular mask dataset provided by Liu et al. [23]. The irregular mask data contains 6 cat-

Table 2
Quantitative comparisons of our method with five advanced inpainting methods on PARIS STREET dataset. + Higher is better. – Lower is better.

	PARIS STREET						
	Masks	1–10%	10–20%	20–30%	30–40%	40–50%	50–60%
PSNR+	PEN	31.230	28.021	26.325	24.326	22.143	20.542
	RN	31.354	28.544	26.004	24.004	22.570	21.073
	RFR	33.967	29.300	26.668	24.698	22.732	20.625
	CTSDG	33.501	30.715	27.678	25.886	23.751	21.478
	LGNet	**34.630**	30.371	28.023	25.927	24.343	21.911
	OURs	34.426	**30.761**	**28.634**	**26.783**	**24.382**	**22.086**
SSIM+	PEN	0.947	0.922	0.876	0.824	0.715	0.605
	RN	0.952	0.924	0.882	0.852	0.788	0.726
	RFR	0.965	0.942	0.894	0.848	0.788	0.706
	CTSDG	0.961	0.953	0.921	0.882	0.822	0.746
	LGNet	**0.980**	0.954	0.919	0.894	0.834	0.761
	OURs	0.976	**0.955**	**0.925**	**0.902**	**0.836**	**0.764**
MSE–	PEN	0.054	0.017	0.025	0.037	0.057	0.078
	RN	0.010	0.018	0.023	0.042	0.053	0.069
	RFR	0.008	0.014	0.025	0.036	0.051	0.075
	CTSDG	0.008	0.012	0.022	0.032	0.049	0.065
	LGNet	**0.005**	0.012	0.020	0.030	0.042	0.062
	OURs	0.006	**0.011**	**0.019**	**0.029**	**0.041**	**0.060**
LPIPS–	PEN	0.056	0.084	0.119	0.184	0.254	0.356
	RN	0.051	0.083	0.124	0.174	0.217	0.372
	RFR	0.035	0.069	0.120	0.223	0.239	0.396
	CTSDG	0.039	0.065	0.118	0.185	0.221	0.342
	LGNet	**0.028**	**0.063**	0.115	0.171	0.214	0.389
	OURs	0.030	**0.063**	**0.107**	**0.156**	**0.206**	**0.307**

Table 3
Quantitative comparisons of our method with five advanced inpainting methods on PLACE2 dataset. + Higher is better. – Lower is better.

	PLACE2						
	Masks	1–10%	10–20%	20–30%	30–40%	40–50%	50–60%
PSNR+	PEN	33.031	27.823	25.428	23.936	22.509	21.159
	RN	34.368	30.115	27.987	26.365	24.229	22.514
	RFR	38.125	32.425	29.330	27.245	25.734	23.096
	CTSDG	34.022	30.194	27.393	25.902	24.569	22.808
	LGNet	38.125	32.425	29.330	27.245	25.734	23.096
	OURs	**38.987**	**32.598**	**29.510**	**27.431**	**25.870**	**23.437**
SSIM+	PEN	0.970	0.926	0.881	0.840	0.794	0.740
	RN	0.973	0.936	0.921	0.878	0.826	0.781
	RFR	0.971	0.929	0.880	0.830	0.780	0.722
	CTSDG	0.972	0.940	0.896	0.863	0.830	0.776
	LGNet	0.988	0.964	0.931	0.897	0.864	0.794
	OURs	**0.990**	**0.969**	**0.938**	**0.898**	**0.865**	**0.803**
MSE–	PEN	0.005	0.014	0.023	0.031	0.041	0.054
	RN	0.005	0.011	0.014	0.023	0.035	0.046
	RFR	0.005	0.013	0.023	0.036	0.050	0.074
	CTSDG	0.004	0.010	0.017	0.024	0.031	0.044
	LGNet	**0.003**	0.007	0.013	0.019	0.025	**0.039**
	OURs	**0.003**	**0.006**	**0.012**	**0.014**	**0.022**	**0.039**
LPIPS–	PEN	0.047	0.108	0.166	0.212	0.261	0.310
	RN	0.047	0.080	0.122	0.104	0.246	0.313
	RFR	0.041	0.091	0.148	0.198	0.251	0.309
	CTSDG	0.043	0.075	0.127	0.158	0.191	0.276
	LGNet	0.011	0.031	0.056	0.081	0.109	**0.170**
	OURs	**0.010**	**0.030**	**0.051**	**0.075**	**0.105**	0.199

egories with different hole ratios, i.e., (0.01, 0.1], (0.1, 0.2], (0.2, 0.3], . . ., (0.5, 0.6]. Each category has 2,000 masks.

Comparison Methods. In this work, we compare our method with five state-of-the-art inpainting methods, which are summarized as follows:

PEN [8]: A pyramid-context encoder network to repair the missing regions in feature map by progressively learning region affinity with attention.

RN [43]: A image inpainting method for computing feature normalization of hole regions and background regions separately in feature map.

RFR [44]: A progressive inpainting method in the feature space with recurrent feature reasoning and knowledge consistent attention.

CTSDG [45]: A two-stream network for image inpainting, which models the structure-constrained texture synthesis and texture-guided structure reconstruction in a coupled manner.

LGNet [36]: A three-stage image inpainting framework to set receptive fields of different sizes to enhance local and global inpainting results respectively.

Implementation Details. Our network is built on the PyTorch framework, trained on NVIDIA 3090 GPU (24 GB) with a batch size of 8, and optimized by the Adam optimizer with a learning rate of 2 × 10–4, where β_1 = 0.5 and β_2 = 0.999. All the masks and images are resized to 256 ×256.

4.2. Performance Comparison with State-of-the-art

Quantitative Comparisons. We conduct quantitative experiments on all datasets with irregular masks, and the mask size accounts for 10% to 60% of the whole image area. In addition, we also perform a center mask inpainting experiment for Celeba dataset. For the evaluation metrics, we adopt several common metrics in the image inpainting task: peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), mean-square error (MSE) and Learned Perceptual Image Patch Similarity (LPIPS). The first two metrics are based on the low-level pixel values, while the last two metrics are related to the high-level visual perception. Their equations are shown below. Where $I(i,j)$ and $K(i,j)$ represent the pixel value of the corresponding position, $\mu_x\mu_y$ represent the aver-

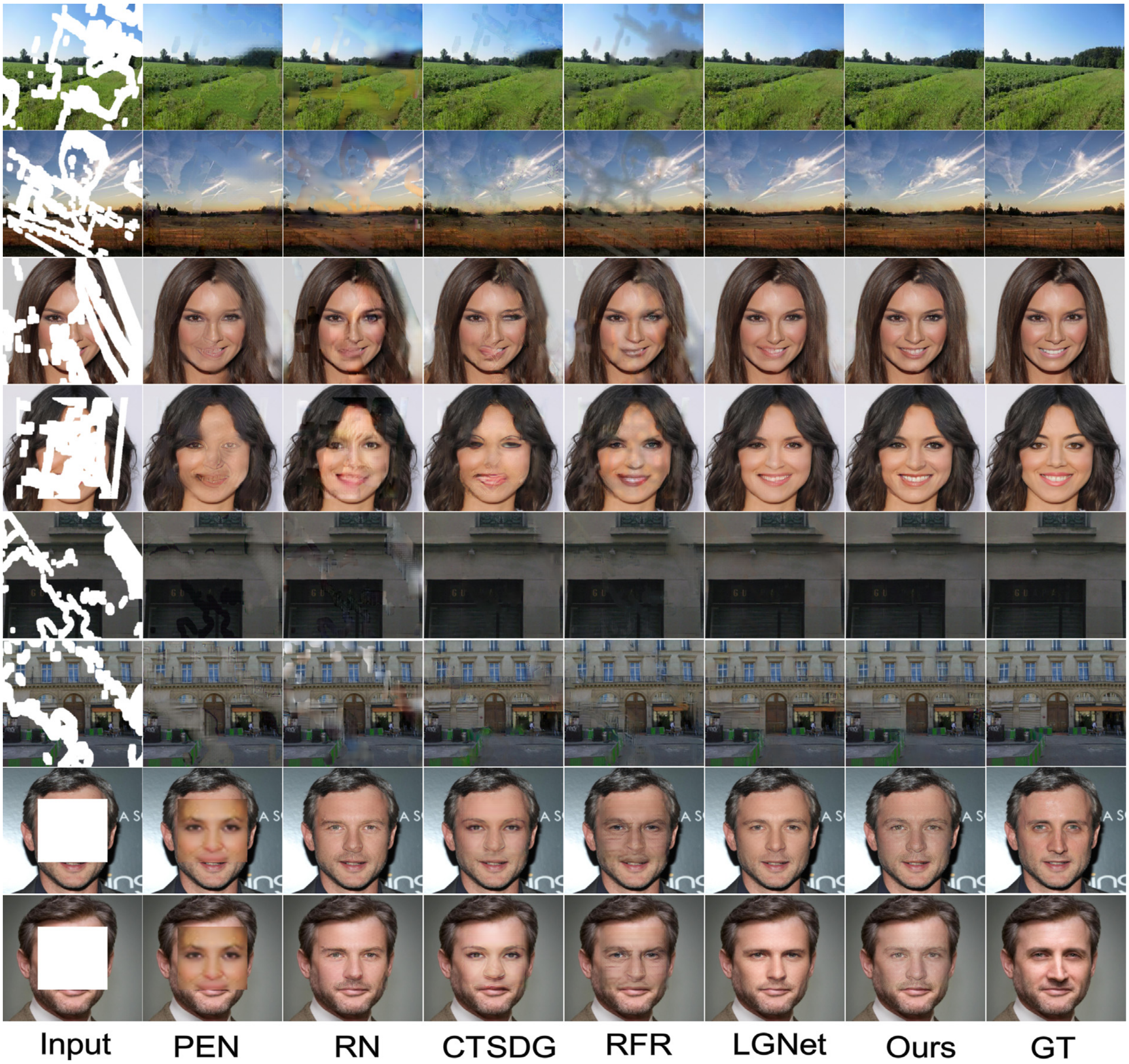


Fig. 6. Qualitative comparisons of our method with PEN, RN, CTSDG, RFR, and LGNet on three datasets with irregular and center masks.

age value of the image, $\sigma_x\sigma_y$ represent the variance of the image, and σ_{xy} is the covariance of two images. Tables 1–3 show the performance of all methods on each dataset. It is obvious that our proposed method outperforms among all comparison methods on these measurements except some indicators on Paris street dataset (When the mask ratio under 10%), but competitive.

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\|I(i,j) - K(i,j)\|^2 \tag{21}$$

$$PSNR = 10 \times \log_{10}\left(\frac{(2^n - 1)^2}{MSE}\right) \tag{22}$$

$$SSIM(x,y) = \frac{\left(2\mu_x\mu_y + c_1\right)(2\sigma_{xy} + c_2)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)} \tag{23}$$

$$LPIPS(x,x_0) = \sum_l \frac{1}{H_l W_l}\sum_{h,w}\|w_l \odot (\hat{y}^l_{hw} - \widehat{y_0}^l_{hw})\|_2^2 \tag{24}$$

Qualitative Comparisons.

We compare our proposed method with five state-of-the-art methods in term of structural and textural coherence. We conduct qualitative experiments on the test set of three datasets with free-form and center masks. As shown in Fig. 6, We take the images with irregular and center mask as the test set. It can be obviously seen that PEN, RN, RFR and CTSDG tend to synthesise distinct unsmooth and blurred final results and LGNet gets relatively realistic results. In contrast, our proposed method can generate smoother inpainting results with reasonable semantics and richer textures with the help of the self-supervised siamese contrastive network and the learning of structure-aware and texture-aware features. It demonstrates that our proposed method is superior to the comparison methods in terms of consistent structures and textures. Furthermore, as shown in the last two lines in Fig. 6, we also conduct experiments on the test images of CelebA-HQ with typical rectangular squares to evaluate the inpainting ability of our proposed method. Our method can generate face images with consistent colors and structures.

4.3. Ablation Studies

We conduct ablation experiments to validate and evaluate our network design by comparing different variants of CSTFNet. The experiments are conducted on test images of Place2 dataset and the corresponding quantitative and qualitative results are shown in Table 4 and Fig.7 respectively.

On Contrastive Learning. The contrastive learning is developed to enhance the ability of the encoder. We remove the siamese network structure from the proposed model to verify the effect of contrastive learning (i.e., only a group of encoder and decoder in the siamese network are used and no contrastive loss is calculated when calculating the loss). To make the comparison more specific, quantitative and qualitative results are given in Table 4 and Fig.7 respectively, which indicate that contrastive learning contributes to the performance gain.

On Coarse Inpainting. The main function of the coarse inpainting network is to fill a rough result for the missing regions first, and provide some useful information for the refinement inpainting network. We get the final result by removing the coarse repair network and directly inputting the masked image to the fine repair network. It can be seen from Table 4 and Fig.7 that the performance of the model without rough network has a certain decline resulting in obvious fuzzy areas in the final results.

On Structure-aware Inpainting Module. Structure-aware inpainting module is mainly to use higher-level feature maps to guide the structure-aware inpainting of its adjacent lower-level feature maps. We removed the structure-aware inpainting module (i.e., only the texture -aware inpainting module is used to repair the feature map) in the ablation experiment to verify its impact on the final results. The quantitative and qualitative results in Table 4 and Fig.7 also verify their necessity.

On Texture-aware Inpainting Module. Texture-aware inpainting module mainly uses low-level feature maps to guide the inpainting of texture-aware of it's adjacent high-level feature maps. Similar to the structure-aware inpainting experiment, we removed the texture-aware inpainting module (only the structure-aware inpainting module is retained when repairing the feature map) in the experiment and verified its irreplaceable value.

On Structure-aware and Texture-aware Features Fusion module (STF). STF is mainly used to fuse the repaired structure-

Table 4
Quantitative results of ablation studies on CELEBA-HQ, PARIS STREET, PLACE2 datasets. + Higher is better. – Lower is better.

	Ablation Learning					
	Masks	10–20%	20–30%	30–40%	40–50%	50–60%
PSNR**+**	wo/Structure	28.147	24.876	24.558	23.895	21.969
	wo/texture	28.323	27.010	25.531	23.637	21.456
	wo/Coarse	29.856	27.792	26.009	24.442	21.735
	wo/Contrastive	30.407	27.515	25.511	24.035	21.809
	wo/Fusion	28.172	27.126	25.498	24.164	20.620
	OURs(full)	**32.598**	**29.510**	**27.431**	**25.870**	**23.437**
SSIM**+**	wo/Structure	0.917	0.847	0.832	0.802	0.729
	wo/texture	0.925	0.901	0.868	0.812	0.733
	wo/Coarse	0.942	0.913	0.875	0.825	0.729
	wo/Contrastive	0.945	0.908	0.864	0.804	0.726
	wo/Fusion	0.916	0.900	0.861	0.816	0.702
	OURs(full)	**0.969**	**0.938**	**0.898**	**0.865**	**0.803**
MSE–	wo/Structure	0.014	0.025	0.031	0.035	0.049
	wo/texture	0.013	0.018	0.025	0.034	0.049
	wo/Coarse	0.010	0.016	0.023	0.032	0.049
	wo/Contrastive	0.010	0.017	0.024	0.035	0.051
	wo/Fusion	0.014	0.018	0.025	0.032	0.056
	OURs(full)	**0.006**	**0.012**	**0.014**	**0.022**	**0.039**
LPIPS–	wo/Structure	0.095	0.170	0.183	0.211	0.266
	wo/texture	0.088	0.106	0.133	0.78	0.240
	wo/Coarse	0.070	0.096	0.130	0.173	0.244
	wo/Contrastive	0.059	0.096	0.134	0.203	0.262
	wo/Fusion	0.094	0.104	0.138	0.177	0.264
	OURs(full)	**0.030**	**0.051**	**0.075**	**0.105**	**0.199**

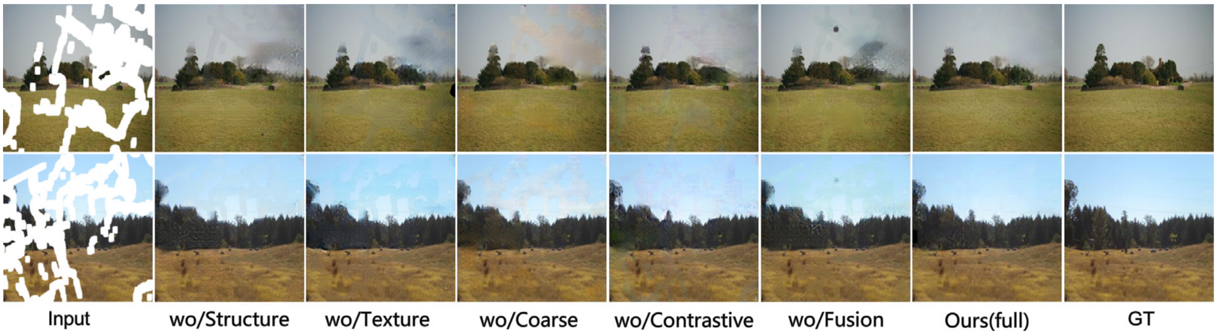


Fig. 7. Visualization of ablation studies on Places2.

aware and texture-aware features reasonably to generate smooth results. In the ablation experiment, we use a simple feature fusion module (a channel-wise concatenation followed by a convolution layer). As shown in Fig.7 and Table 4, we demonstrate that STF obviously benefits the quality of the results, with consistent textures and better quantitative scores reported.

5. Conclusion

In this paper, a novel two-stage and two-stream image inpainting method is proposed to generate reasonable and visually realistic images. To be exact, the proposed method boosts the ability of feature extraction of encoder by settings of contrastive learning. Then, we design a structure to repair each layer of feature map from the perspective of structure-aware and texture-aware, and followed by a fusion module to naturally fuse the two repaired features. Experiments on CelebA-HQ, Pairs Street and Places2 show the superiority of our proposed method in generating smoother, coherent and rich textures results.

CRediT Author Contribution statement

Long Chen: Conceptualization, Data curation, Writing-original, Methodology, Investigation. **Changan Yuan:** Funding acquisition, Supervision. **Xiao Qin:** Formal analysis, Validation. **Wei Sun:** Data curation, Writing-review & editing. **Xiaofeng Zhu:** Project administration, Supervision

Data availability

The data that has been used is confidential.

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.

Acknowledgements

This work was supported by the National Key Research and Development Program of China (Grant No. 2022YFD2401200), the Key Project of Science and Technology of Guangxi (Grant No. AA22068057), and the National Natural Science Foundation of China (Grant No. 61962006).

References

[1] C. Barnes, E. Shechtman, A. Finkelstein, D.B. Goldman, Patchmatch: A randomized correspondence algorithm for structural image editing, ACM Trans. Graph. 28 (3) (2009) 24.
[2] K.H. Jin, J.C. Ye, Annihilating filter-based low-rank hankel matrix approach for image inpainting, IEEE Transactions on Image Processing 24 (11) (2015) 3498–3511.
[3] D. Jin, C. Huo, C. Liang, L. Yang, Heterogeneous graph neural network via attribute completion, in: Proceedings of the Web Conference 2021, 2021, pp. 391–400.
[4] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, A.A. Efros, Context encoders: Feature learning by inpainting, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2536–2544.
[5] S. Iizuka, E. Simo-Serra, H. Ishikawa, Globally and locally consistent image completion, ACM Transactions on Graphics (ToG) 36 (4) (2017) 1–14.
[6] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T.S. Huang, Generative image inpainting with contextual attention, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 5505–5514.
[7] K. Nazeri, E. Ng, T. Joseph, F.Z. Qureshi, M. Ebrahimi, Edgeconnect: Generative image inpainting with adversarial edge learning, arXiv preprint arXiv:1901.00212 (2019).
[8] Y. Zeng, J. Fu, H. Chao, B. Guo, Learning pyramid-context encoder network for high-quality image inpainting, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 1486–1494.
[9] H. Liu, B. Jiang, Y. Xiao, C. Yang, Coherent semantic attention for image inpainting, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 4170–4179.
[10] Z. Yu, D. Jin, Z. Liu, D. He, X. Wang, H. Tong, J. Han, As-gcn: Adaptive semantic architecture of graph convolutional networks for text-rich networks, in: 2021 IEEE International Conference on Data Mining (ICDM), IEEE, 2021, pp. 837–846.
[11] D. Jin, Z. Yu, P. Jiao, S. Pan, D. He, J. Wu, P. Yu, W. Zhang, A survey of community detection approaches: From statistical modeling to deep learning, IEEE Transactions on Knowledge and Data Engineering (2021).
[12] J. Gan, R. Hu, Y. Mo, Z. Kang, L. Peng, Y. Zhu, X. Zhu, Multi-graph fusion for dynamic graph convolutional network, IEEE Transactions on Neural Networks and Learning Systems (2022), 10.1109/TNNLS.2022.3172588.
[13] L. Peng, R. Hu, F. Kong, J. Gan, Y. Mo, X. Shi, X. Zhu, Reverse graph learning for graph neural network, IEEE Transactions on Neural Networks and Learning Systems (2022), 10.1109/TNNLS.2022.3161030.
[14] C. Yuan, Z. Zhong, C. Lei, X. Zhu, R. Hu, Adaptive reverse graph learning for robust subspace learning, Information Processing & Management 58 (6) (2021).
[15] Y. Zhu, J. Ma, C. Yuan, X. Zhu, Interpretable learning based dynamic graph convolutional networks for alzheimer's disease analysis, Information Fusion 77 (2022) 53–61.
[16] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, arXiv preprint arXiv:1511.07122 (2015).
[17] T. Ruzic, A. Pizurica, Context-aware patch-based image inpainting using markov random field modeling, IEEE transactions on image processing 24 (1) (2015) 444–456.
[18] N. Kawai, T. Sato, N. Yokoya, Diminished reality based on image inpainting considering background geometry, IEEE transactions on visualization and computer graphics 22 (3) (2015) 1236–1247.
[19] D. Ding, S. Ram, J.J. Rodríguez, Image inpainting using nonlocal texture matching and nonlinear filtering, IEEE Transactions on Image Processing 28 (4) (2018) 1705–1719.
[20] Q. Sun, R. Zhai, F. Zuo, Y. Zhong, Y. Zhang, A review of image inpainting automation based on deep learning, in: Journal of Physics: Conference Series, Vol. 2203, IOP Publishing, 2022, p. 012037.
[21] K. Li, Y. Wei, Z. Yang, W. Wei, Image inpainting algorithm based on tv model and evolutionary algorithm, Soft Computing 20 (3) (2016) 885–893.
[22] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Communications of the ACM 63 (11) (2020) 139–144.
[23] G. Liu, F.A. Reda, K.J. Shih, T.-C. Wang, A. Tao, B. Catanzaro, Image inpainting for irregular holes using partial convolutions, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 85–100.
[24] H. Liu, B. Jiang, Y. Song, W. Huang, C. Yang, Rethinking image inpainting via a mutual encoder-decoder with feature equalizations, in: European Conference on Computer Vision, Springer, 2020, pp. 725–741.
[25] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T.S. Huang, Free-form image inpainting with gated convolution, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 4471–4480.

[26] L. Peng, Y. Mo, J. Xu, J. Shen, X. Shi, X. Li, H.T. Shen, X. Zhu, Grlc:graph representation learning with constraints, IEEE Transactions on Neural Networks and Learning Systems (2022), https://doi.org/10.1109/TNNLS.2022.3230979.

[27] Z. Wu, Y. Xiong, S.X. Yu, D. Lin, Unsupervised feature learning via non-parametric instance discrimination, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 3733–3742.

[28] Y. Tian, D. Krishnan, P. Isola, Contrastive multiview coding, in: European conference on computer vision, Springer, 2020, pp. 776–794.

[29] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in: International conference on machine learning, PMLR, 2020, pp. 1597–1607.

[30] K. He, H. Fan, Y. Wu, S. Xie, R. Girshick, Momentum contrast for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 9729–9738.

[31] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, A. Joulin, Unsupervised learning of visual features by contrasting cluster assignments, Advances in Neural Information Processing Systems 33 (2020) 9912–9924.

[32] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, et al., Bootstrap your own latent-a new approach to self-supervised learning, Advances in neural information processing systems 33 (2020) 21271–21284.

[33] X. Chen, K. He, Exploring simple siamese representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 15750–15758.

[34] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer, 2015, pp. 234–241.

[35] X. Mao, Q. Li, H. Xie, R.Y. Lau, Z. Wang, S. Paul Smolley, Least squares generative adversarial networks, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2794–2802.

[36] W. Quan, R. Zhang, Y. Zhang, Z. Li, J. Wang, D.-M. Yan, Image inpainting with local and global refinement, IEEE Transactions on Image Processing 31 (2022) 2405–2420.

[37] S. Woo, J. Park, J.-Y. Lee, I.S. Kweon, Cbam: Convolutional block attention module, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 3–19.

[38] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: European conference on computer vision, Springer, 2016, pp. 694–711.

[39] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, A. Torralba, Places: A 10 million image database for scene recognition, IEEE transactions on pattern analysis and machine intelligence 40 (6) (2017) 1452–1464.

[40] T. Karras, T. Aila, S. Laine, J. Lehtinen, Progressive growing of gans for improved quality, stability, and variation, arXiv preprint arXiv:1710.10196 (2017).

[41] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 3730–3738.

[42] C. Doersch, S. Singh, A. Gupta, J. Sivic, A. Efros, What makes paris look like paris?, ACM Transactions on Graphics 31 (4) (2012) 101:1–101:9.

[43] T. Yu, Z. Guo, X. Jin, S. Wu, Z. Chen, W. Li, Z. Zhang, S. Liu, Region normalization for image inpainting, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 12733–12740.

[44] J. Li, N. Wang, L. Zhang, B. Du, D. Tao, Recurrent feature reasoning for image inpainting, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 7760–7768.

[45] X. Guo, H. Yang, D. Huang, Image inpainting via conditional texture and structure dual generation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 14134–14143.

Long Chen received the B.S. degree from Anhui University of Science and Technology, in 2016, and the master's degree from Nannning Normal University in 2020. he is currently pursuing the Ph.D. degree with UESTC (University of Electronic Science and Technology of China). His research interests include machine learning, computer vision and graph learning.

Chang-An Yuan is currently a professor at the Guangxi Academy of Sciences. He received the Ph.D. degree in computer application technology from Sichuan University, China, in 2006. He is currently a member of database Committee of Chinese Computer Society, a member of Machine Learning Committee of Chinese Association for Artificial Intelligence, and the president of Guangxi Association for Artificial Intelligence. His currently research interests include computational intelligence and data mining.

Xiao Qin is currently a professor at the School of Computer and Information Engineering, Nanning Normal University. She received the M.S. degree in computer application technology from Nanning Normal University, China, in 2009. She is a master's tutor of computer Science and Technology in Nanning Normal University. Her currently research interests include machine learning and computer vision.

Sun Wei, born in Hunan, China, in 1987, studied at Chengdu University of Technology from 2006 to 2010 and received a bachelor's degree in 2010, studied at Southwest Jiaotong University from 2010 to 2014 and received a master's degree in 2014. Studied at Southwest University from 2015 to 2018 and received a PhD in 2018, worked as a postdoctoral fellow at Shenzhen University in 2020, and became an associate professor at Guangxi Academy of Sciences in 2022. Her research interests include Network and New Media.

Xiaofeng Zhu received the PhD degree in computer science from The University of Queensland, Australia. He has authored or coauthored about 120 peer-reviewed research papers, such as the IEEE Transactions on Image Processing, the IEEE Transactions on Neural Networks and Learning Systems, the IEEE Transactions on Knowledge and Data Engineering, ACM Multimedia, Conference on Computer Vision and Pattern Recognition, Association for the Advancement of Artificial Intelligence, The International Joint Conference on Artificial Intelligence, and International Conference on Medical Image Computing and Computer Assisted Intervention. His research interests include machine learning and image analysis. Specifically, he is focusing on mining useful knowledge or information from big multimedia or medical imaging data.