Pattern Recognition 119 (2021) 108036

Contents lists available at ScienceDirect

Pattern Recognition

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

Feature learning and patch matching for diverse image inpainting

Yuan Zeng [a,*], Yi Gong [b,*], Jin Zhang [c]

[a] Academy for Advanced Interdisciplinary Studies, Southern University of Science and Technology (SUSTech), Shenzhen 518055, PR China
[b] University Key Laboratory of Advanced Wireless Communications of Guangdong Province, Southern University of Science and Technology, Shenzhen 518055, PR China
[c] Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, PR China

ARTICLE INFO

Article history:
Received 28 September 2020
Revised 30 March 2021
Accepted 10 May 2021
Available online 29 May 2021

Keywords:
Diverse image inpainting
Free-form mask
U-Net-like network
Nearest neighbors

ABSTRACT

We present an image inpainting approach to generate diverse high-quality inpainting results. Recent advances in deep adversarial networks have led to significant improvements in the challenging task of filling large holes in natural images. Although deep generative models can generate visually plausible structures and textures, most of them are not interpretable, making it difficult to control the inpainting output. In addition, deep generative models do not have capacity to produce diverse results for each input. To address such limitations, we design a novel free-form image inpainting framework with two sequential steps: the first step formulates the inpainting process as a regression problem and utilizes a U-Net-like convolutional neural network to map an input to a coarse inpainting output, and the second step utilizes the nearest neighbor based pixel-wise matching to map the coarse output to diverse high-quality outputs. The second step allows our approach to compose novel high-quality content by copy-pasting high-frequency missing information from different training exemplars. Experiments on multiple datasets, i.e., CelebA-HQ, AFHQ, and Paris StreetView, show that our approach is able to offer multiple natural outputs with higher diversity in a controllable manner.

1. Introduction

Image inpainting is the task of filling holes in images, which is an active research topic in computer vision and serves various applications, such as object removal, image-based blending and image denoising. The key challenge of image inpainting lies in synthesizing both global semantic visual perceptions and local textured patterns that are coherent with background regions. Given images with holes in Fig. 1, what would our humans imagine to be occupying these holes? Although we may universally agree on high-level semantics, we can easily imagine multiple variations of eyebrows, eyes, and mouths that could be plausible and pleasing to the eye. Based on this observation, this work focuses on synthesizing diverse plausible results, which is different from approaches that attempt to generate only a single result.

Early image inpainting approaches attempted to solve the problem using ideas similar to exemplar-based texture synthesis [1], i.e., by matching and copying background patches into holes starting from low-resolution to high-resolution or propagating from hole boundaries. Though these approaches work well for synthesizing texture-consistent outputs [2,3], they cannot generate semantically meaningful contents. More recently, some learning-based image inpainting approaches [4,5] were proposed to treat image inpainting as a conditional generation problem and infer semantic content using deep generative adversarial networks (GAN) [6]. Although these approaches can learn semantics from large scale datasets and synthesize novel content in an end-to-end fashion, they have two limitations. First, given an image with holes, humans can imagine multiple plausible inpainting results while these approaches are limited to generate only one result. Second, the learning models are difficult to explain or interpret, making the inpainting output difficult to control or modify.

To generate diverse outputs, conditional variational autoencoders[7] are utilized in some methods [8–10], where a low-dimensional latent code sampled from the standard Gaussian distribution is injected into the network to generate multiple outputs. However, these methods only consider a mapping between two domains and require to train various generators to obtain diverse inpainting results based on a single ground truth, which help prevent to scale with the increasing numbers of domains, and produce results with limited diversity.

To address these limitations, we propose a free-form image inpainting framework using a classic learning architecture that can naturally allow for multiple outputs. An important insight we will

* Corresponding authors.
E-mail addresses: zengy3@sustech.edu.cn (Y. Zeng), gongy@sustech.edu.cn (Y. Gong).

https://doi.org/10.1016/j.patcog.2021.108036
0031-3203/

Fig. 1. Sampled results of our image completion approach. From left to right, input image with holes, automatic restoration using our deep generative model in stage I, exemplar 1, our result based on exemplar 1, exemplar 2, our result based on exemplar 2, exemplar 3, and our result based on exemplar 3.

use is that the nearest neighbor based pixel-wise matching, as a non-parametric learning method, can match a low-resolution input query to a large corpus of training pairs and return a corresponding high-resolution output. This provides a solution to the problem of having a single output per trained model. The proposed image inpainting framework consists of two-stages: a learning based coarse image completion and a fine texture synthesis. We first formulate the free-form image inpainting as a regression problem and make use of a U-Net-like network for coarse image completion. U-Net architecture [11] is a well-known convolutional neural network (CNN) for pixel-wise image generation since it propagates context information to higher resolution layers with a large number of feature channels in the upsampling part. U-Net model with regression loss is a one-to-one mapping and tends to generate a smooth output that looks like a smoothed average of all potential outputs. In the second stage, we present a nearest neighbor based compositional matching to further improve the diversity and quality of the regressed output. Since our outputs are obtained by copy-pasting high-frequency missing information from training exemplars, we can efficiently match to a large number of training exemplars in a controllable manner. As illustrated in Fig. 1, using compositional matching is critical for generating diverse high-quality inpainting results. Our main contributions are summarized as follows:

- A two-stage exemplar-guide image inpainting framework is introduced for free-form image completion. It is able to generate diverse high-quality inpainting results in a controllable manner.
- We design a U-Net-like generative model trained with a joint loss, including per-pixel losses and neural feature based losses, for free-form image inpainting. No matter whether missing regions are irregular or centering, our generative model can synthesize novel content with better color and structure consistency.
- We present a nearest neighbor based compositional matching for diverse image inpainting. Different from the nearest neighbor based patch-matching in existing image inpainting algorithms, where high-quality patches are directly used, we adopt compositional matching to copy-past high-frequency missing information from multiple training exemplars. It is simple, interactive, and produces diverse high-quality inpainting results with less generated artifacts.
- We demonstrate that the proposed framework can generate higher-quality inpainting results than recent state of the arts on challenge inpainting datasets, including CelebA-HQ [12,13], AFHQ [14], and Paris StreetView [15].

The reminder of this article is organized as follows. Related work is given in Section 2. In Section 3, we describe the proposed free-form image inpainting framework in detail. In Section 4, we introduce the experimental setup and illustrate the image inpainting performance of the proposed framework. Finally, in Section 5, conclusions are drawn.

2. Related work

2.1. Image inpainting

A variety of approaches have been proposed for image inpainting. Traditional diffusion-based approaches [16,17] typically use variational algorithms or patch similarity to propagate information from the background regions into the missing regions. Although these approaches work well on small or narrow missing regions, they often fail on large missing regions. Compared with diffusion-based approaches, patch-based approaches can fill in large missing regions using texture synthesis techniques. Patch-based image completion was first proposed in [18], where neighboring patches from a source image are copied and pasted into the target image. Criminisi et al. [1] proposed to optimize patch search using multiple scales and orientations. However, computing patch similarity for every target-source pair is computationally expensive. To address this challenge, an approximate nearest neighbor algorithm, called PatchMatch [19], was proposed for real-time patch matching and image inpainting. Later, Korman et al. [20] proposed coherency sensitive hashing to further improve search speed and ac-

curacy. Ding et al. [21] proposed an exemplar-based image inpainting approach to inpaint geometrical structures and textures well. Since these methods only use low-level features for patch matching, they cannot fill in holes with semantic or novel content.

Recently, deep learning based approaches have emerged as a promising paradigm for image inpinting. Initially, deep CNNs were trained for inpainting of small and thin missing regions [22]. Context Encoder [4] first proposed GAN-based model for inpainting of large holes, where an encoder-decoder network was trained to handle 64×64-sized holes. Yang et al. [23] proposed a high resolution image inpainting approach using multi-scale neural patch synthesis. Iizuka et al. [24] proposed a globally and locally consistent image completion approach, where global discriminator and local discriminator were introduced to determine generation consistency of the output and Poisson blending was used as a post-processing to enforce color coherency. In [25], Yu et al. proposed a unified feed-forward generative network with a contextual attention layer. In [26], Zeng et al. proposed a controllable image inpainting framework by adopting an end-to-end deep generative model and a nearest neighbor based global matching. However, this approach is mainly trained on a large centering square mask and does not generalize well on masks of arbitrary shape, size, and location. To better handle irregular masks, partial convolution [27] was introduced for image inpainting, where the convolution was masked and re-normalized to utilize valid pixels only. In [28], an image inpainting method using attention mechanism and partial convolution was proposed to obtain more realistic inpainting results. Zheng et al. [10] presented a pluralistic image inpainting approach to generate diverse inpainting results. In [5], a generative image inpainting system based on gated convolutions and a patch-based GAN loss was proposed for free-form image inpainting.

2.2. Interpretable and conditional image generation

Very recently, a few generative models [29] have been proposed to address the problem of image inpainting under conditions. Lee et al. [30] proposed a diverse image to image translation approach using disentangled representations, where limited style codes were extracted from an encoder network. This limitation comes from the fact that the learning-based generation models use pre-determined labels and thus inevitably produce the same output per each domain. Unlike deep learning based approaches, exemplar-based approaches fill holes by copy-pasting similar patches from the known regions. Thus, their outputs are more interpretable. To tackle the problem of generating novel content, Hays and Efros [31] proposed an image inpainting approach which searches neighboring patches in 2 million training images. Later, Whyte et al. [32] extended the approach in [31] to a particular case where images with the same scene are included in the dataset. Zhu et al. [33] presented a user-guide image editing approach to control the image editing process. Ding et al. [21] proposed a perceptually aware image inpainting approach that works well on geometrical structures and textures. Instead of using a predefined set of editing operations, Zeng et al. [26] proposed an image restoration framework for diverse image inpainting. However, this approach can only generate limited diverse outputs by doing global matching on training exemplars, and it produces visible artifacts with limited exemplars. To produce a large set of diverse high-quality inpainting results, our compositional matching based image inpainting allows users to have arbitrarily fine control of the final outputs. Specifically, unlike global matching where nearest neighbors were searched in one exemplar, compositional matching searches nearest neighbors from multiple exemplars to achieve much higher diversity. The inpainting process in this work can explicitly reveal how each pixel in output is generated, such as recover the eye using the eye from this training image and recover the mouth using the mouth from another training image, see Fig. 2. In addition, this process makes our approach quite interpretable, where a user is able to control the output via editing exemplars.

2.3. Pixel-wise correspondence

Generating pixel-wise correspondences between missing regions and training exemplars is a key issue for the nearest neighbor based compositional matching. It has been one of the core challenges in computer vision to establish such pixel-wise correspondences. Early studies on pixel-wise correspondence are generally based on hand-crafted features. Recently, CNN based approaches were proposed to learn correspondences between images. Long et al. [34] first proposed to use neural features from a pre-trained CNN to establish pixel-wise correspondences. To further improve the correspondences, the following works focus on incorporating additional annotations [35,36], and adopting a coarse-to-fine strategy [37]. Inspired by feature extraction in style transfer, where multi-scale neural features can capture both high-level semantics and low-level details for better pixel representation, we use multi-scale neural features to establish the pixel-wise correspondences. Such pixel-wise correspondences enable us to extract high-frequency missing information from training exemplars to synthesize high-quality outputs.

3. Approach

Given an input image with free-form holes, we aim to fill the holes with diverse semantic content. In this section, we formulate this problem as a conditional image inpainting problem, where the conditions are the exemplars, and introduce a two-stage framework for diverse image inpainting. Fig. 3 shows an overview of our framework. In stage I, a U-Net-like network $\mathcal{F}_{\mathbf{w}}$ is designed to transform an input image x with a free-form mask into a single inpainting output $\hat{y}$ via $\hat{y} = \mathcal{F}_{\mathbf{w}}(x)$. The network $\mathcal{F}_{\mathbf{w}}$ is trained to minimize a weighted combination of loss functions, that is,

$$\widetilde{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmin}} E_{x,y}\left(\sum_i \lambda_i \mathcal{L}_i(\mathcal{F}_{\mathbf{w}}(x), y)\right), \tag{1}$$

where $\mathcal{L}_i(\mathcal{F}_{\mathbf{w}}(x), y)$ is the ith loss function that measures similarity between the output $\mathcal{F}_{\mathbf{w}}(x)$ and the ground truth y. The single output $\hat{y}$ is a coarse estimation of the ground truth y and looks like a smoothed average of all the potential outputs that could be generated. To obtain multiple high-quality outputs, we use the nearest neighbor based compositional reconstruction to copy-past high-frequency missing information from neighboring exemplars in stage II. To consider both local textures and global semantics in patch matching, we draw inspiration from recent works that generate images via optimization [38,39]. The key insight of these approaches is that different layers of a deep CNN tend to capture different spatial information. Specifically, multi-scale features from a pre-trained VGG-16 network $\mathcal{V}$ are used to build pixel-wise correspondences between the network output and similarly-smoothed training exemplars. Since the training exemplars used in stage II can be selected by users and the nearest neighbor based matching is interactive, we can efficiently generate a large number of exemplar-based high-frequency outputs in a controllable manner.

3.1. Learning model based one-to-one mapping

U-Net based image inpainting network Inspired by the U-Net based regression models in image inpainting of irregular holes, our image inpainting network in stage I follows a U-Net-like architecture [11] similar to the one used in [29]. The input of the network is a $256 \times 256 \times 3$ RGB image with free-form holes, and the output

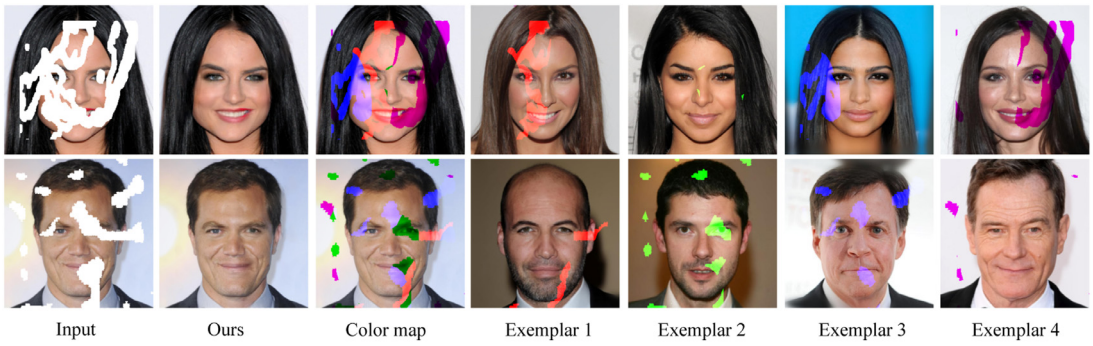

Fig. 2. Illustration of compositional reconstruction. Given an input image with free-form missing regions, we show high-frequency result obtained by doing compositional matching on 4 neighboring training exemplars. The correspondences associated with the 4 neighboring exemplars are illustrated using color code pixels.

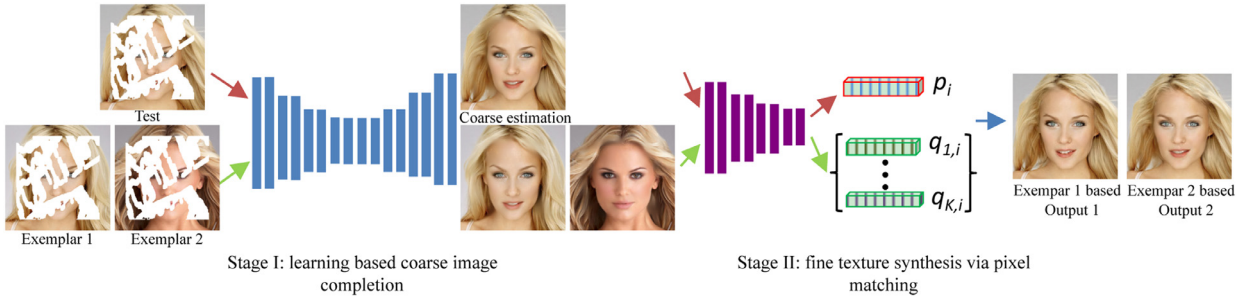

Fig. 3. Overview of our framework for diverse image inpainting. In stage I, a U-Net-like network is deigned to roughly fill in the missing regions. Then conditioned on the coarse output and training exemplars, we establish pixel-wise correspondences between the output and similarly-smoothed exemplars using multi-scale nueral features, and perform nearest neighbor based compositional matching to obtain diverse high-quality inpainting outputs in stage II.

is a $256 \times 256 \times 3$ RGB image with entire content. The encoder extracts the features of the input and reduces its spatial dimension, and the decoder restores the feature dimensions and generates an inpainting output. Specifically, the encoder consists of 8 double convolutional layers, and the decoder consists of 7 double up-convolutional layers [40]. Each double convolutional layer is composed of a 3×3 convolution and a 4×4 dilated convolution. The 3×3 convolutions double the number of channels and keep the same spatial size, and the 4×4 dilated convolutions keep the same channel number and reduce the spatial size by half. Each convolution in the encoder is followed by batch normalization and Leaky Rectified Linear Unit (ReLU), and each convolution in the decoder is followed by batch normalization and ReLU. The encoder and decoder are connected by a channel-wise fully connected layer, and the last layer is a convolutional layer with 3 channels output. The skip links are used to concatenate two feature maps from an encoder and its mirrored decoder, acting as the feature inputs for the next convolution layer.

Network architecture We present the architecture details of our U-Net-like image inpainting network in Table 1. Specifically, Convi_j indicates a convolutional layer with specified filter size, number of channels, stride, and padding. Each Conv$i$ consists of two convolutional layers. DeConvi_j indicates a de-convolutional layer. In addition, Conv1–8 are in the encoder stage, and DeConv1–8 are in the decoder stage. The BatNorm column indicates whether Conv layer is followed by a Batch Normalization layer, and the Nonlinearity shows whether or what nonlinearity function is used after Batch Normalization. Concat indicates skip links, which concatenate the previous results with the corresponding Conv results from the encoder stage.

3.2. Loss functions

We train our U-Net-like network by regressing the output $\hat{y} = \mathcal{F}_{\mathbf{w}}(x)$ to the ground truth y. However, there are often multiple equally plausible ways to fill holes. To handle both high-level semantic and low-level texture in the inpainting network, we train the network with a decoupled joint loss function. We use per-pixel losses to handle per-pixel reconstruction accuracy. In addition, to address the shortcoming of per-pixel losses and allow our loss functions to better measure perceptual and semantic differences between images, we make use of the VGG-16 network $\mathcal{V}$ [41]. The network $\mathcal{V}$ has been pre-trained for image classification on ImageNet [14] as a fixed loss network to define our perceptual and semantic loss functions. This sub-section describes different components of our loss function.

Per-pixel losses Let $\hat{y}_i$ denote the output of the inpainting network for a given input x_i, and y_i denote the ground truth of the input x_i. We use the l_1 distance to define our per-pixel losses, that are,

$$\mathcal{L}_m = \left\| M \odot (\hat{y}_i - y_i) \right\|_1, \tag{2}$$

and

$$\mathcal{L}_b = \left\| (1 - M) \odot (\hat{y}_i - y_i) \right\|_1, \tag{3}$$

where $\odot$ is the pixel-wise multiplication and M is a binary mask with a value of 1 inside the missing regions and 0 otherwise. The mask M makes the losses to be computed on the missing regions and the known regions separately, which are denoted as $\mathcal{L}_m$ and $\mathcal{L}_b$ respectively. Note that the input image can be expressed as $x_i = (1 - M) \odot y_i$.

Perceptual loss Let $\mathcal{V}_j(x_i)$ be the activations of the jth layer of the network $\mathcal{V}$ for the input x_i, which is a feature map of shape $C_j \times H_j \times W_j$. Similar as the perceptual loss introduced in [42], our perceptual loss function $\mathcal{L}_{per}$ is defined as

$$\mathcal{L}_{per} = E\left[\sum_j \left\| \mathcal{V}_j(\hat{y}_i) - \mathcal{V}_j(y_i) \right\|_1 + \sum_j \left\| \mathcal{V}_j(y_{\mathrm{com},i}) - \mathcal{V}_j(y_i) \right\|_1 \right]. \tag{4}$$

Table 1
Image inpainting network architecture.

Module Name	Filter size	Channels	Stride	Padding	BatNorm	Nonlinearity
$Conv1_{1,2}$	$3\times3, 4\times4$	64	1,2	1	Y	LeakyReLU(0.2)
$Conv2_{1,2}$	$3\times3, 4\times4$	64	1,2	1	Y	LeakyReLU(0.2)
$Conv3_{1,2}$	$3\times3, 4\times4$	128	1,2	1	Y	LeakyReLU(0.2)
$Conv4_{1,2}$	$3\times3, 4\times4$	128	1,2	1	Y	LeakyReLU(0.2)
$Conv5_{1,2}$	$3\times3, 4\times4$	256	1,2	1	Y	LeakyReLU(0.2)
$Conv6_{1,2}$	$3\times3, 4\times4$	256	1,2	1	Y	LeakyReLU(0.2)
$Conv7_{1,2}$	$3\times3, 4\times4$	512	1,2	1	Y	LeakyReLU(0.2)
$Conv8_{1,2}$	$3\times3, 4\times4$	512	1,2	1	Y	LeakyReLU(0.2)
$DeConv1_1$	3×3	4000	1	1	Y	LeakyReLU(0.2)
$DeConv1_2$	4×4	512	2	1	Y	ReLU
Concat1 (Conv7)		512+512				
$DeConv2_{1,2}$	$3\times3, 4\times4$	512	1,2	1	Y	ReLU
Concat2 (Conv6)		256+512				
$DeConv3_{1,2}$	$3\times3, 4\times4$	256	1,2	1	Y	ReLU
Concat3 (Conv5)		256+256				
$DeConv4_{1,2}$	$3\times3, 4\times4$	256	1,2	1	Y	ReLU
Concat4 (Conv4)		128+256				
$DeConv5_{1,2}$	$3\times3, 4\times4$	128	1,2	1	Y	ReLU
Concat5 (Conv3)		128+128				
$DeConv6_{1,2}$	$3\times3, 4\times4$	128	1,2	1	Y	ReLU
Concat6 (Conv2)		64+128				
$DeConv7_{1,2}$	$3\times3, 4\times4$	64	1,2	1	Y	ReLU
Concat7 (Conv1)		64+64				
$DeConv8_{1,2}$	$3\times3, 4\times4$	64	1,2	1	Y	ReLU
Conv	3×3	3	1	0	-	-

Specifically, our perceptual loss function is based on feature information from $relu1_1$ and $relu1_2$ ($J=2$) for the pre-trained VGG-16 network, and we use it to penalize both raw output $\hat{y}_i$ and composited output $y_{com,i}=M\odot\hat{y}_i+x_i$ when they are not perceptually similar to the target y_i.

Style loss As demonstrated in [43], including style loss is able to combat checkerboard artifacts caused by transpose convolution layers. We use style loss in [39] to further penalize differences in style. Our style loss is defined as

$$\mathcal{L}_{sty}=E\left[\sum_j\left\|G_{\mathcal{V}_j}(\hat{y}_i)-G_{\mathcal{V}_j}(y_i)\right\|_1+\sum_j\left\|G_{\mathcal{V}_j}(y_{\mathrm{com},i})-G_{\mathcal{V}_j}(y_i)\right\|_1\right], \tag{5}$$

where $G_{\mathcal{V}_j}$ is a $C_j\times C_j$ Gram matrix, and it is constructed from a feature map $\mathcal{V}_j$. We use feature maps from $relu1_1$ and $relu1_2$ in our style loss.

Total variation loss We also add a total variation (TV) loss $\mathcal{L}_{tv}$ to encourage smoothness, which is defined as

$$\mathcal{L}_{tv}=\sum_{m,n}\left(\left\|y^{m+1,n}_{\mathrm{com},i}-y^{m,n}_{\mathrm{com},i}\right\|_2^2+\left\|y^{m+1,n}_{\mathrm{com},i}-y^{m,n}_{\mathrm{com},i}\right\|_2^2\right). \tag{6}$$

Joint loss We define the overall loss function as

$$\mathcal{L}=\lambda_{mis}\mathcal{L}_m+\lambda_{bac}\mathcal{L}_b+\lambda_{per}\mathcal{L}_{per}+\lambda_{sty}\mathcal{L}_{sty}+\lambda_{tv}\mathcal{L}_{tv}. \tag{7}$$

where λ_{mis}, λ_{bac}, λ_{per}, λ_{sty} and λ_{tv} are scales.

3.3. Nearest neighbor based one-to-many mapping

Global matching Given a test image y_t, we aim to refine the coarse estimation $\hat{y}_t$ and generate diverse high-quality outputs by copy-pasting high-frequency information from training exemplars. To generate diverse outputs, we make use of the K-nearest-neighbor (KNN) algorithm, a classic non-parametric approach to obtain K outputs. Unlike conventional exemplar-based image inpinting, where low resolution images are updated by directly copy-pasting nearest neighbors from high-resolution exemplars, we address the problem of high resolution missing information by copy-pasting similar information from training exemplars. Given an exemplar y_e, we can use it to predict the high frequency missing information of the inpainting network $\mathcal{F}_{\mathbf{w}}$, and generate a final output $\bar{y}_t$ as

$$\bar{y}_{t,p}=\hat{y}_{t,p}+(y_{e,q}-\hat{y}_{e,q}), \tag{8}$$

where $\hat{y}_{t,p}$ denotes pixel p in the intermediate output $\hat{y}_t$, and $\hat{y}_{e,q}$ denotes pixel q in the similar intermediate output $\hat{y}_e$, and $\bar{y}_t$ denotes our final output. The pixel q is searched as

$$q=\underset{n}{\operatorname{argmin}}\,S(\hat{y}_{t,p},\hat{y}_{e,n}), \tag{9}$$

where $S(\hat{y}_{t,p},\hat{y}_{e,n})$ denotes pixel-wise similarity between a query pixel $\hat{y}_{t,p}$ and an exemplar pixel $\hat{y}_{e,n}$ with pixel index $n=\{1,\cdots,N\}$. We measure the pixel-wise similarity by computing cosine distance between two pixel descriptors, as described in subsection distance function.

Compositional matching Although global matching can synthesize different outputs using different exemplars, it is limited to the number of high-quality images in the training set. We propose a compositional matching to synthesize a much larger set of outputs by matching a query pixel to pixels in multiple exemplars. Rather than using KNN for global reconstruction, we refine the intermediate output $\hat{y}_t$ in pixel-wise as

$$\bar{y}_{t,p}=\hat{y}_{t,p}+(y_{k,q}-\hat{y}_{k,q}), \tag{10}$$

where $y_{k,q}$ is pixel q in a training example y_k, and $\hat{y}_{k,q}$ is pixel q in the intermediate output $\hat{y}_k$. The pixel q is then searched as

$$(k,q)=\underset{m,n}{\operatorname{argmin}}\,S(\hat{y}_{t,p},\hat{y}_{m,n}), \tag{11}$$

where $S(\hat{y}_{t,p},\hat{y}_{m,n})$ denotes distance function measuring similarity between two pixels $\hat{y}_{t,p}$ and $\hat{y}_{m,n}$ with image index $m=\{1,\cdots,M\}$.

Distance function A critical issue of the nearest neighbor based patch matching is the choice of distance function S. Considering both local and global information in distance function can significantly improve image generation performance [24,44]. Inspired by recent works that the different layers of a pre-trained deep network can capture the different amount of spatial context [42], we extract multi-scale feature maps from different layers of VGG-16 network, which has been pre-trained on ImageNet [14] for image classification. The weights of the pre-trained network are directly loaded and fixed for feature extraction. VGG-16 network has

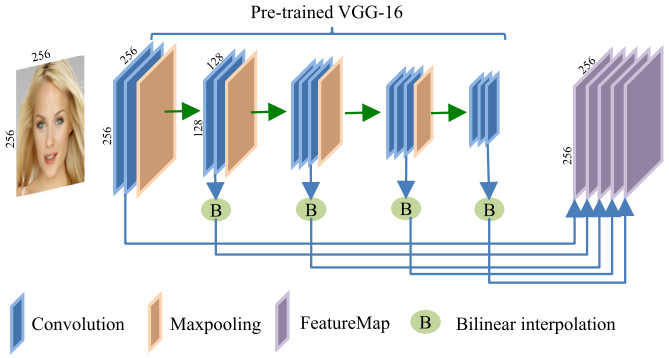

Fig. 4. Illustration of constructing pixel-wise neural feature representation via the pre-trained VGG-16.

Fig. 5. Sampled masks of arbitrary missing regions with rectangular blocks, brush drawing holes and irregular holes.

5 scales with 13 convolutional layers denoted as $conv1_1$, $conv1_2$, $conv2_1$, $conv2_2$, $conv3_1$, $conv3_2$, $conv3_3$, $conv4_1$, $conv4_2$, $conv4_3$, $conv5_1$, $conv5_2$, and $conv5_3$. We use features from $conv1_2$, $conv2_2$, $conv3_3$, $conv4_3$, and $conv5_3$ to construct a pixel descriptor. The channel size of the 5 layers are 64, 128, 256, 512, 512, respectively.

Next, we merge feature maps from the 5 different convolutional layers to construct pixel descriptors. As shown in Fig. 4, since the size of feature maps from different layers is ordinarily different, we use bilinear interpolation to up-sample each feature map to the size of the output. Given an input image with the size of $M \times N$, the size of our pixel descriptors is $M \times N \times T$ ($T = 64 + 128 + 256 + 512 + 512 = 1472$). Let us consider a problem of matching a query pixel p in an intermediate output $\hat{y}_t$ to pixels in exemplars. We extract patches with size 3×3 centered on pixel p from all up-sampled feature maps and build a feature cube with size of $3 \times 3 \times T$. After that, we reshape the feature cube into a matrix with dimension of $3^2 \times T$, and denote the matrix as $\mathbf{h}_p$. To find the closest-matching pixel q in M intermediate reconstructions of training exemplars, we measure pixel similarity $S(p, q)$ with normalized inner product (cosine similarity) between two descriptors $\mathbf{h}_p$ and $\mathbf{h}_q$, that is

$$S(p,q) = \left\langle \frac{\mathbf{h}_p}{\|\mathbf{h}_p\|}, \frac{\mathbf{h}_q}{\|\mathbf{h}_q\|} \right\rangle. \tag{12}$$

Exemplar search for efficient match Although inpainting performance of the nearest neighbor based compositional matching is proportional to the number of training images, the computational complexity of our pixel-wise matching would vary linearly with the size of training exemplars if we directly search for every pixel in training set. The computational efficiency can be improved by first reporting back the K best global matches. The number of training exemplars used for the compositional matching is then reduced from M to K ($K << M$). In addition, similar to most exemplar-based image inpainting approaches, the inpainting performance of our approach is diminished by the textural and structural information difference between exemplars and the input image. Thus, it is natural to select exemplars with similar textural and structural information to the input image. To select K closest-matching exemplars from the training set, we do KNN on image descriptors. Specifically, we use features from the convolutional layer $conv5_4$ for the pre-trained VGG-16 to construct the image descriptors, and compute inner product between image descriptors to select top-K closest-matching exemplars for compositional matching. Since each output of the stage II is computed by copy-pasting high-frequency information from nearest neighbors, we can generate diverse plausible inpainting outputs by varying the number of closest-matching exemplars.

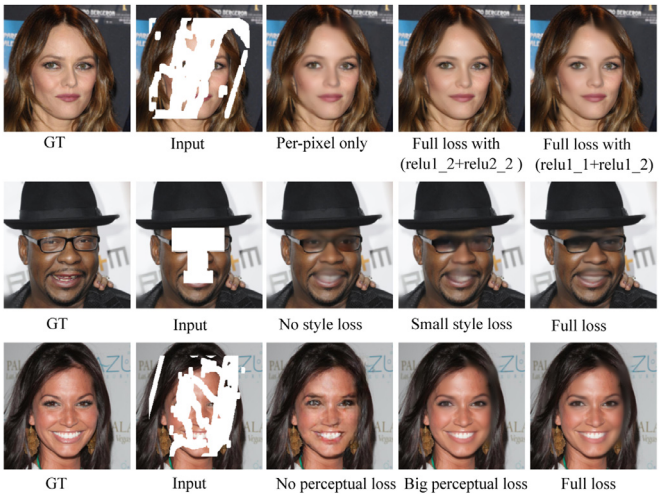

Fig. 6. Image inpainting using different losses for network training. The inpainting result with just per-pixel losses only are well aligned, but not sharp. Using full loss with $relu1_2$ and $relu2_2$, results are sharper than using full loss with $relu1_1$ and $relu1_2$, but have more artifacts on complex content (e.g. eyes and mouths). Removing perceptual loss term from the total loss, results are sharp but not well coherent. Using the total loss with a small style loss, results are not sharp. In addtion, removing style loss term from the total loss or using a large perceptual loss weight in the total loss, results are sharp but contain checkerboard artifacts.

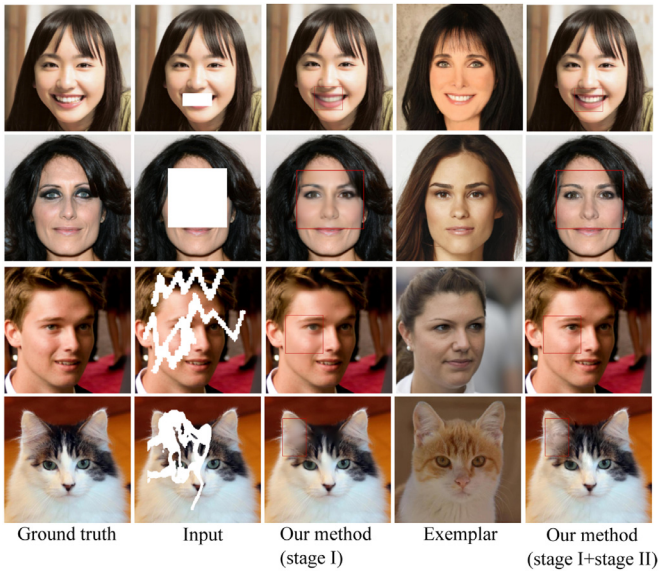

Fig. 7. Comparison of our final outputs with the intermediate outputs from the inpainting network on CelebA-HQ faces and AFHQ. From left to right, ground truth, input image with holes, intermediate inpainting output, exemplar and exemplar-based final inpainting result. We observe that the nearest neighbor based patch matching can copy-past similar missed high-frequency information from exemplars and generate high quality images.

Fig. 8. Visual comparison between global matching and compositional matching. From left to right, intermediate result of our inpainting network, global matching based final result, and compositional matching based final result.

Fig. 9. Controllable pluralistic image inpainting on CelebA-HQ faces. Our approach can generate pluralistic inpainting results based on an exemplar.

4. Experiments

4.1. Dataset

We evaluate the proposed image inpainting framework on CelebA-HQ [12,13], AFHQ [45] and Paris StreetView [15]. The CelebA-HQ dataset consists of 30,000 high quality face images with size of 1024×1024. We randomly select 29,000 images for training and use remaining 1,000 images for test. To compare our approach with exiting image inpainting algorithms, we also use train and val splits in [5] for CelebA-HQ. The Animal Faces-HQ (AFHQ) dataset [45] consists of 15,000 high quality images at 512×512 resolution. We use the original train and val splits in [45] for AFHQ. In addition, we use the original train and val splits in [4] for Paris StreetView [15], which consists of 14,900 training images and 100 validation images.

4.2. Missing region masks

In our experiments, we use three approaches to generate our mask dataset. The first one is loading a fixed irregular mask dataset [27], where masks are collected from an occlusion estimation method between two consecutive frames of videos. The second one is a free-form mask generation method in [5], which aims to generate brush drawing holes, and the last approach is generating rectangular block holes in arbitrary places. Sampled masks of arbitrary missing regions are shown in Fig. 5.

4.3. Implementation details

The proposed framework is implemented in PyTorch and trained on a PC with two NVIDIA TITAN X GPUs. The missing regions in input images are initialized with a constant mean value.

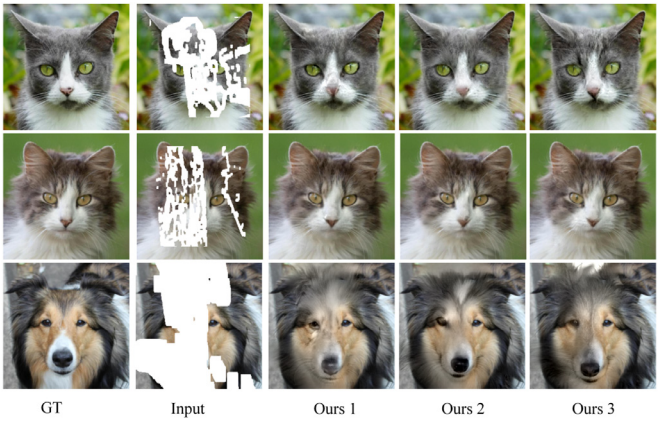

Fig. 10. Controllable pluralistic image inpainting on AFHQ. Our approach generates multiple inpainting results of animal faces.

The image inpainting network is trained with a batch size of 32 and optimized with Adam optimizer [46]. The learning rate starts with 0.0002 and decays 0.999 every epoch. The training process is terminated when the validation loss is not decreased within 30 epochs, and the model with the smallest validation loss is saved for testing. The training of CelebA-HQ face inpainting model takes two days, whereas the training of AFHQ animal face inpainting model or Paris StreetView model takes one day. After receiving a coarse intermediate output from the trained inpainting model, we use it as an input image of the stage II, and generate diverse inpainting results via varying the number of neighboring exemplars. The nearest neighbor based patch matching is performed on an Intel Core i7-5960X CPU with 8 cores and takes around one second for a 256×256 image.

4.4. Experiment setup

We use four experiments to analyze the inpainting performance of the proposed approach. First, we study the effect of different loss terms on inpainting outputs. Second, we compare our final outputs (from stage II) with our intermediate outputs (from stage I) to evaluate the high-frequency information compensation of stage II, and compare global matching with compositional matching. Later, we evaluate the diversity of our final results via varying exemplars. After that, we compare our method with three existing inpainting methods:

-PM: PatchMatch [19], a fast approximate nearest neighbor based patch matching method for image inpainting, which is known as the state-of-the-art non-learning based inpainting method.

-PC: Partial Conv [27], a partial convolution based deep image inpainting model for irregular holes.

-GC: Gated Conv [5], a deep generative model with gated convolutions and spectral-normalized markovian discriminator for free-form image inpainting.

4.5. Experiment results

Ablation study of different loss terms We do ablation studies to illustrate the effectiveness of different loss terms and the total loss. The hyper-parameters λ_{mis}, λ_{bac}, λ_{per}, λ_{sty} and λ_{tv} are initialized based on the related work [27] and [42], and determined by performing multiple experiments and evaluating reconstruction performance on 100 validation images. We use $\lambda_{mis} = 10$, $\lambda_{bac} = 1$,

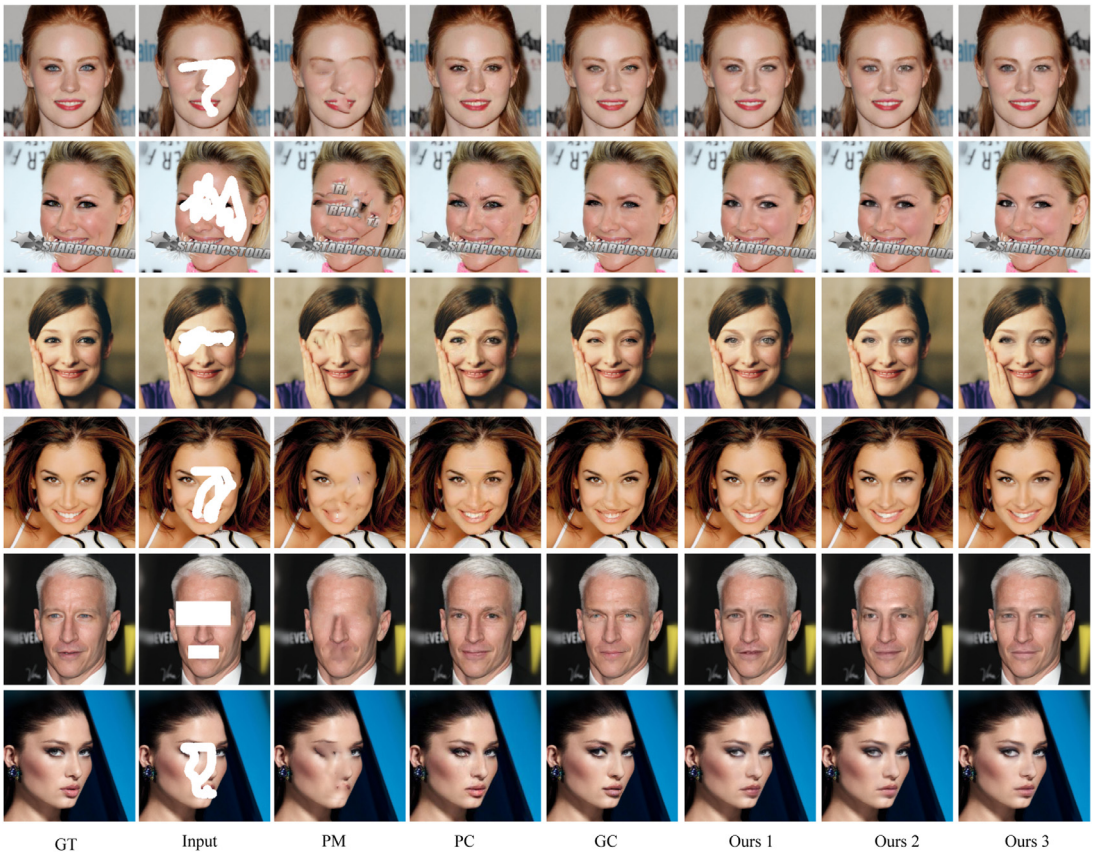

Fig. 11. Qualitative comparisons on CelebA-HQ dataset. From left to right, ground truth, input image with hole, PM [19], PC [27] and GC [5], Ours 1, Ours 2 and Ours 3.

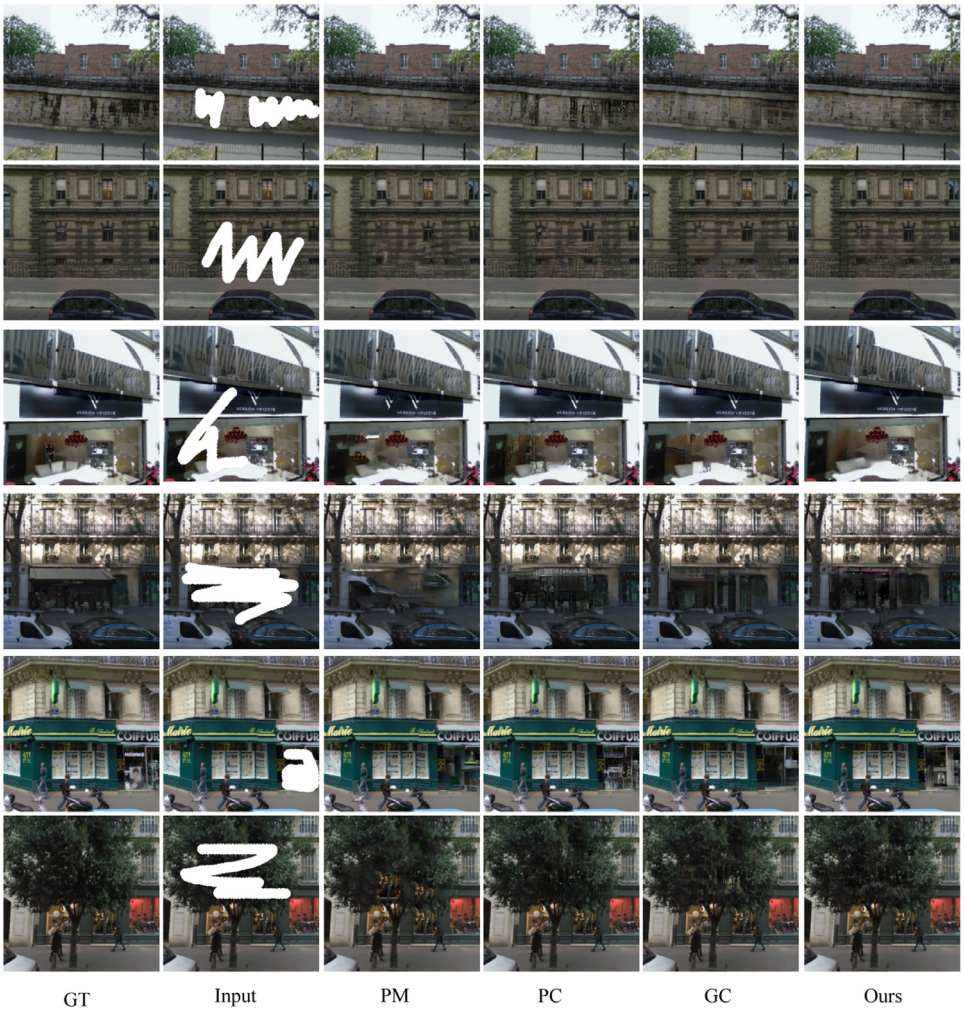

Fig. 12. Qualitative comparisons on Paris StreetView dataset. From left to right, ground truth, input image with hole, PM [19], PC [27] and GC [5], Ours.

$\lambda_{per} = 0.04$, $\lambda_{sty} = 50$ and $\lambda_{tv} = 0.5$. It is known that perceptual loss terms not only generate super-resolution details, but also generate checkerboard artifacts. We exploit the insight from previous works that the effect of checkerboard artifacts will be increased when features from higher layers of the pre-trained VGG-16 network are used in loss functions. Fig. 6 shows inpainting results of the inpainting model trained with different loss terms, which are per-pixel losses only, total loss using features extracted from $relu1_2$ and $relu2_2$, total loss using features extracted from $relu1_1$ and $relu1_2$, removing style loss from total loss, total loss $\mathcal{L}$ with a small style loss weight, removing perceptual loss from total loss, total loss with a large perceptual loss weight and full loss. Compared to the results of the models trained with per-pixel losses only, no style loss or no perceptual loss, the model trained with full loss provides outputs with fewer artifacts. In addition, the model trained with $relu1_2$ and $relu2_2$ based joint loss generates more checkerboard artifacts than the model trained with $relu1_1$ and $relu1_2$ based joint loss.

Low-resolution to high-resolution We compare our final outputs with our intermediate outputs to demonstrate the effectiveness of the nearest neighbor based high-frequency information compensation. The experiment results are shown in Fig. 7. We observe that the coarse inpainting network generates mid-frequencies fairly well, but fails to return plausible high-frequency content. In addition, the comparison results indicate that the nearest neighbor based patch matching can leverage high-frequency information from exemplars and consequently generate results with high-frequency textures.

Global matching vs compositional matching We compare the outputs of the global matching with those of the compositional matching. Visual comparison results are shown in Fig. 8. We observe that the generated outputs of the compositional matching are more natural than those of the global matching. This can be explained that multiple exemplars contain more suitable nearest neighbors to extract the high-frequency information from.

Diverse image inpainting Moreover, experiment results in Fig. 9 and Fig. 10 show that our approach can generate diverse plausible inpainting results via varying exemplars. Instead of matching to the entire training set, a user can manually control the inpainting process via specifying exemplars and instruct the nearest neighbor based one-to-may mapping process to generate a result that looks similar to the exemplars.

Qualitative comparison We first compare our approach with three previous state-of-the-art methods: PM [19], PC [27] and GC [5] on CelebA-HQ. For the comparison, we use content aware fill in Photoshop to generate the inpainting results of PM, and use the provided model to generate the face results of GC. The PC model is trained using the same dataset setting of GC and the training is stopped when the validation loss converges. Fig. 11 shows the visually comparison results on CelebA-HQ. We observe that PM fails to generate plausible content when the holes are large or contain semantic information, since PM is based on locally matched

Table 2
Results of mean l_1 loss, mean l_2 loss and PSNR on validation images of CelebA-HQ with free-form masks.

	PM	PC	GC	ours
l_1 loss	2.78%	0.77%	0.97%	**0.59%**
l_2 loss	0.88%	0.15%	0.18%	**0.10%**
PSNR (dB)	27.47	34.87	33.99	**36.66**

Fig. 13. Inpainting results of our approach when there are no suitable nearest neighbors in exemplars. From left to right, input image with holes, our result and ground truth.

patches. Although PC can generate plausible content, many areas are blurry. The PC can also effectively deal with irregular missing regions composed of texture. However, the details in the restored regions are not as delicate as the background, such as details in eyes and mouths. The deep adversarial model GC can recover irregular holes with plausible content, but it still exhibits observable unpleasant boundaries and artifacts. Since our method is able to copy-past high-frequency missing information from exemplars, it produces various plausible results, which are more visually pleasant with fine details.

In addition, we compare our method with PM, PC and GC on Paris StreetView dataset [15]. Qualitative comparison results are shown in Fig. 12. As shown in the figure, our approach with feature learning and patch matching generates more realistic results with much fewer artifacts than the other three inpainting methods.

Quantitative comparison As mentioned in [47], there is no good quantitative evaluation metric to evaluate image inpainting results. Nevertheless, we report in Table 2 our evaluation results in terms of mean l_1 error, mean l_2 error and peak signal-to-noise ratio (PSNR) on validation images of CelebA-HQ with free-form masks. As shown in Table 2, learning-based methods perform better than PM in terms of mean l_1, mean l_2, and PSNR. In addition, the proposed approach obtains better numerical performance than PC and GC, indicating our inpainting results are closer to the ground truth.

Failure cases Although our approach can generate diverse high-frequency inpainting outputs, it mostly fails when there are no suitable nearest neighbors to extract high-frequency information from, see Fig. 13. One way to deal with this problem is to do an exhaustive pixel-wise nearest neighboring search with more training exemplars and computation time.

5. Conclusions

In this work, we proposed a novel free-form image inpainting framework. The proposed framework is based on the fact that the image inpainting can be structured as a regression problem. We first use a U-Net-like learning model in combination with a joint loss function, including per-pixel losses, perceptual loss, style loss and total variation loss for coarse image completion. The U-Net-like architecture with a large number of feature channels in the upsampling part allows the network to propagate context information to higher resolution layers, and generates more accurate fine-grained details. However, due to the fact that the U-Net-like model directly regresses an output from an incomplete input, the single output tends to look like a smoothed average of all potential outputs that could be generated. We therefore presented a nearest neighbor based patch matching approach to obtain diverse high-frequency inpainting outputs.

The proposed image inpainting framework is analyzed and visually compared with the U-Net-like generative model on CelebA-HQ, AFHQ and Paris StreetView datasets. Experiment results of the U-Net-like generative model showed that the resolution of the outputs is affected by different loss terms, and the resolution of the intermediate outputs is lower than those of the final outputs. In addition, experiment results showed that the outputs of the proposed compositional matching are more natural than those of the existing global matching for free-form image inpainting. Moreover, experiments on quantitative and qualitative comparisons between the proposed approach and the state-of-the-art learning based free-form image inpainting approaches in referenced literature illustrated the superior performance of the proposed inpainting framework.

The proposed framework can generate diverse high-quality inpainting results, which has an important value for exemplar-guide image inpainting. However, the proposed approach may create artifacts when there are no suitable nearest neighbors in training exemplars. To solve this problem, system-level optimization can be made to improve the computational efficiency for doing exhaustive pixel-wise compositional matching. In contrast to other end-to-end learning based free-form image inpainting approaches, the proposed framework consists of two stages, which makes use of the nearest neighbor based compositional matching as a simple post-processing after the U-Net-like generative model. Future research will be interesting to investigate how to include nearest neighbor based compositional matching in deep generative model for end-to-end high-frequency free-form image inpainting while still preserving the interaction of the generation process and the diversity of the inpainting outputs.

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.

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.

Acknowledgments

This research is supported in part by National Key R&D Program of China under Grant 2019YFB1802800, Guangdong Science and Technology Program under Grant 2019A1515110479, Guangdong Basic and Applied Basic Research Foundation under Grant 2019B1515130003, Education Commission of Guangdong under Grant 2020ZDZX3057 and 2019KQNCX128, Shenzhen Science and Technology Research Project under Grant JCYJ20180507181527806.

References

[1] A. Criminisi, P. Perez, K. Toyama, Region filling and object removal by exemplar-based image inpainting, IEEE Trans. Image Process. 13 (9) (2004) 1200–1212.
[2] Y. Zeng, Y. Gong, Nearest neighbor based digital restoration of damaged ancient chinese paintings, 2018 IEEE 23rd International Conference on Digital Signal Processing (2018) 1–5.
[3] S.-S. Wang, S.-L. Tsai, Automatic image authentication and recovery using fractal code embedding and image inpainting, Pattern Recognit 41 (2) (2008) 701–712.
[4] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell, A. Efros, Context encoders: Feature learning by inpainting, in: Proceedings of the IEEE Computer Vision and Pattern Recognition, 2016.
[5] 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 International Conference on Computer Vision, 2019.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems 27, Curran Associates, Inc., 2014, pp. 2672–2680.
[7] D.P. Kingma, M. Welling, An introduction to variational autoencoders, Found. Trends Mach. Learn. 12 (4) (2019) 307–392.
[8] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, J. Choo, StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018.
[9] P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.
[10] C. Zheng, T.-J. Cham, J. Cai, Pluralistic image completion, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 1438–1447.
[11] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention, in: LNCS, volume 9351, Springer, 2015, pp. 234–241.
[12] 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.
[13] T. Karras, T. Aila, S. Laine, J. Lehtinen, Progressive growing of GANs for improved quality, stability, and variation, in: International Conference on Learning Representations, 2018.
[14] R. Olga, D. Jia, S. Hao, K. Jonathan, S. Sanjeev, M. Sean, H. Zhiheng, K. Andrej, K. Aditya, B. Michael, A.C. Berg, L. Fei-Fei, ImageNet large scale visual recognition challenge, Int J Comput Vis 115 (3) (2015) 211–252.
[15] C. Doersch, S. Singh, A. Gupta, J. Sivic, A.A. Efros, What makes paris look like paris? ACM Transactions on Graphics (SIGGRAPH) 31 (4) (2012) 1–9.
[16] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, J. Verdera, Filling-in by joint interpolation of vector fields and gray levels, IEEE Trans. Image Process. 10 (8) (2001) 1200–1211.
[17] Levin, Zomet, Weiss, Learning how to inpaint from global image statistics, in: Proceedings of the IEEE International Conference on Computer Vision, 2003, pp. 305–312 vol.1.
[18] A.A. Efros, T.K. Leung, Texture synthesis by non-parametric sampling, in: Proceedings of the IEEE International Conference on Computer Vision, 1999.
[19] C. Barnes, E. Shechtman, A. Finkelstein, D.B. Goldman, Patchmatch: a randomized correspondence algorithm for structural image editing, ACM Trans Graph 28 (3) (2009) 1–11.
[20] S. Korman, S. Avidan, Coherency sensitive hashing, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE Computer Society, 2011, pp. 1607–1614.
[21] D. Ding, S. Ram, J.J. Rodriguez, Perceptually aware image inpainting, Pattern Recognit 83 (2018) 174–184.
[22] J. Xie, L. Xu, E. Chen, Image denoising and inpainting with deep neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 341–349.
[23] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, H. Li, High-resolution image inpainting using multi-scale neural patch synthesis, in: Proceedings of the IEEE Computer Vision and Pattern Recognition, 2017.
[24] S. Iizuka, E. Simo-Serra, H. Ishikawa, Globally and locally consistent image completion, ACM Trans Graph 36 (4) (2017) 1–14.
[25] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T.S. Huang, Generative image inpainting with contextual attention, in: Proceedings of the IEEE Computer Vision and Pattern Recognition, IEEE Computer Society, 2018, pp. 5505–5514.
[26] Y. Zeng, Y. Gong, X. Zeng, Controllable digital restoration of ancient paintings using convolutional neural network and nearest neighbor, Pattern Recognit Lett 133 (2020) 158–164.
[27] 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, 2018.
[28] N. Wang, S. Ma, J. Li, Y. Zhang, L. Zhang, Multistage attention network for image inpainting, Pattern Recognit 106 (2020) 107448.
[29] P. Isola, J. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.
[30] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, M.-H. Yang, Diverse image-to-image translation via disentangled representations, in: Proceedings of the European Conference on Computer Vision, Springer, 2018.
[31] J. Hays, A.A. Efros, Scene completion using millions of photographs, ACM Trans Graph 26 (3) (2007).
[32] O. Whyte, J. Sivic, A. Zisserman, Get out of my picture! internet-based inpainting, BMVC, British Machine Vision Association, 2009.
[33] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, A.A. Efros, Generative visual manipulation on the natural image manifold, in: Proceedings of the European Conference on Computer Vision, Springer, 2016.
[34] J.L. Long, N. Zhang, T. Darrell, Do convnets learn correspondence? in: Advances in Neural Information Processing Systems 27, Curran Associates, Inc., 2014, pp. 1601–1609.
[35] A. Kanazawa, D.W. Jacobs, M. Chandraker, Warpnet: Weakly supervised matching for single-view reconstruction, in: Proceedings of the IEEE Computer Vision and Pattern Recognition, IEEE Computer Society, 2016, pp. 3253–3261.
[36] T. Zhou, S. Tulsiani, W. Sun, J. Malik, A.A. Efros, View synthesis by appearance flow, in: Proceedings of the European Conference on Computer Vision, in: Lecture Notes in Computer Science, volume 9908, Springer, 2016, pp. 286–301.
[37] J. Liao, Y. Yao, L. Yuan, G. Hua, S.B. Kang, Visual attribute transfer through deep image analogy, ACM Trans Graph 36 (4) (2017).
[38] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, H. Lipson, Understanding neural networks through deep visualization, in: Deep Learning Workshop, International Conference on Machine Learning, 2015.
[39] L.A. Gatys, A.S. Ecker, M. Bethge, Image style transfer using convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.
[40] E. Shelhamer, J. Long, T. Darrell, Fully convolutional networks for semantic segmentation, IEEE Trans Pattern Anal Mach Intell 39 (4) (2017) 640–651.
[41] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations, 2015.
[42] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: Proceedings of the European Conference on Computer Vision, 2016, pp. 694–711.
[43] M.S.M. Sajjadi, B. Schölkopf, M. Hirsch, EnhanceNet: Single image super-resolution through automated texture synthesis, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE, 2017, pp. 4501–4510.
[44] B. Hariharan, P. Arbelaez, R. Girshick, J. Malik, Hypercolumns for object segmentation and fine-grained localization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.
[45] Y. Choi, Y. Uh, J. Yoo, J.-W. Ha, StarGAN v2: Diverse image synthesis for multiple domains, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2020.
[46] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: International Conference on Learning Representations, 2015.
[47] J. Sun, L. Yuan, J. Jia, H.-Y. Shum, Image completion with structure propagation, in: ACM Transactions on Graphics, 2005, pp. 861–868.

Yuan Zeng is an assistant professor at Southern University of Science and Technology, Shenzhen, China. She obtained her Ph.D. degree in Signal and Information Processing from Delft University of Technology, the Netherlands. Her main research interests are intelligent signal processing, including speech enhancement and image restoration.

Yi Gong is a professor at Southern University of Science and Technology, Shenzhen, China. He obtained his Ph.D degree in electrical engineering from Hong Kong University of Science and Technology. His principal research interest s are on topics related to intelligent communication systems and signal processing

Jin Zhang is an assistant professor at Southern University of Science and Technology, Shenzhen, China. She obtained her Ph.D. de gree in computer science from Hong Kong University of Science and Technology. Her main research interests are mobile computing and collaborative communication.