VCNet: A Robust Approach to Blind Image Inpainting

Yi Wang[1], Ying-Cong Chen[2], Xin Tao[3], and Jiaya Jia[1,4]

[1]The Chinese University of Hong Kong [2]MIT CSAIL [3]Kuaishou Technology
[4]SmartMore
{yiwang, leojia}@cse.cuhk.edu.hk
ycchen@csail.mit.edu jiangsutx@gmail.com

Fig. 1: Our blind inpainting method on raindrop removal (**left**, from [29]), face (**top right**, from FFHQ [17]), and animal (**bottom right**, from ImageNet [6]). *No* masks are provided during inference, and these filling patterns are not included in our training.

Abstract. Blind inpainting is a task to automatically complete visual contents without specifying masks for missing areas in an image. Previous work assumes known missing-region-pattern, limiting the application scope. We instead relax the assumption by defining a new blind inpainting setting, making training a neural system robust against various unknown missing region patterns. Specifically, we propose a two-stage visual consistency network (VCN) to estimate where to fill (via masks) and generate what to fill. In this procedure, the unavoidable potential mask prediction errors lead to severe artifacts in the subsequent repairing. To address it, our VCN predicts semantically inconsistent regions first, making mask prediction more tractable. Then it repairs these estimated missing regions using a new spatial normalization, making VCN robust to mask prediction errors. Semantically convincing and visually compelling content can be generated. Extensive experiments show that our method is effective and robust in blind image inpainting. And our VCN allows for a wide spectrum of applications.

Keywords: Blind image inpainting · visual consistency · spatial normalization · generative adversarial networks

1 Introduction

Image inpainting aims to complete missing regions of an image based on its context. Generally, it takes a corrupted image as well as a mask that indicates

missing pixels as input, and restore it based on the semantics and textures of uncorrupted regions. It serves applications of object removal, image restoration, etc. We note the requirement of having accurate masks makes it difficult to be practical in several scenarios where masks are not available, *e.g.*, graffiti and raindrop removal (Fig. 1). Users need to carefully locate corrupted regions manually, where inaccurate masks may lead to inferior results. We in this paper analyze blind inpainting that automatically finds pixels to complete, and propose a suitable solution based on image context understanding.

Existing work [3, 24] on blind inpainting assumes that the missing areas are filled with constant values or Gaussian noise. Thus the corrupted areas can be identified easily and almost perfectly based on noise patterns. This oversimplified assumption could be problematic when corrupted areas are with unknown content. To improve the applicability, we relax the assumption and propose the *versatile blind inpainting* task. We solve it by taking deeper semantics of the input image into overall consideration and detecting more semantically meaningful *inconsistency* based on the context in contrast to previous blind inpainting.

Note that blind inpainting without assuming the damage patterns is highly ill-posed. This is because the unknown degraded regions need to be located based on their difference from the intact ones instead of their known characteristics, and the uncertainties in this prediction make the further inpainting challenging. We address it in two aspects, i.e., a new data generation approach and a novel network architecture.

For training data collection, if we only take common black or noise pixels in damaged areas as input, the network may detect these patterns as features instead of utilizing the contextual semantics as we need. In this scenario, the damage for training should be diverse and complicated enough so that the contextual inconsistency instead of the pattern in damage can be extracted. Our first contribution, therefore, is the new strategy to generate diverse training data where natural images are adopted as the filling content with random strokes.

For model design, our framework consists of two stages of mask prediction and robust inpainting. A discriminative model is used to conduct binary pixel-wise classification to predict inconsistent areas. With the mask estimated, we use it to guide the inpainting process. Though this framework is intuitive, its specific designs to address the biggest issue in this framework are non-trivial: *how to neutralize the generation degradation brought by inevitable mask estimation errors* in the first stage. To cope with this challenge, we propose a probabilistic context normalization (PCN) to spatially transfers contextual information in different neural layers, enhancing information aggregation of the inpainting network based on the mask prediction probabilities. We experimentally validate that it outperforms other existing approaches exploiting masks, e.g., concatenating mask with the input image and using convolution variants (like Partial Convolution [22] or Gated Convolution [44]) to employ masks, in evaluation.

Though trained without seeing any graffiti or trivial noise patterns (*e.g.* constant color or Gaussian noise), our model can automatically remove them without manually annotated marks, even for complex damages introduced by real im-

ages. This is validated in several benchmarks like FFHQ [17], ImageNet [6], and Places2 [49]. Besides, we find our predicted mask satisfyingly focuses on visual inconsistency in images as expected instead of inherent damage patterns when these two stages are jointly trained in an adversarial manner. This further improves robustness for this very challenging task, and leads to the application of exemplar-guided face-swap (Sec. 4.3). Also, such blind inpainting ability can be transferred to other removal tasks such as severe raindrop removal as exemplified in Fig. 1 and Sec. 4.2. Many applications are enabled.

Our contribution is twofold. First, we propose the first relativistic generalized blind inpainting system. It is robust against various unseen degradation patterns and mask prediction errors. We jointly model mask estimation and inpainting procedure, and address error propagation from the computed masks to the subsequent inpainting via new spatial normalization. Second, effective tailored training data synthesis for this new task is presented with comprehensive analysis. It makes our blind inpainting system robust to visual inconsistency, which is beneficial for various inpainting tasks.

2 Related Work

Blind Image Inpainting Conventional inpainting methods employ external or internal image local information to fill missing regions [2, 4, 5, 14, 18, 19, 32]. For the blind image setting, existing research [3, 24, 7, 41, 36, 47] assumes contamination with simple data distributions, *e.g.* text-shaped or thin stroke masks filled with constant values. This setting makes even a simple model applicable by only considering local information, without understanding the semantics of the input.

Generative Image Inpainting Recent advancement [1, 26, 46, 34] in the conditional generative models makes it possible to fill large missing areas in images [28, 13, 20, 42, 43, 37, 40, 39, 48, 31, 38, 45, 39, 48, 35, 23]. Pathak *et al.* [28] learned an inpainting encoder-decoder network using both reconstruction and adversarial losses. Iizuka *et al.* [13] proposed the global and local discriminators for the adversarial training scheme. To obtain more vivid texture, coarse-to-fine [42, 43, 40] or multi-branch [37] network architecture, and non-local patch-match-like losses [42, 37] or network layer [43] were introduced.

Specifically, Yang *et al.* [42] applied style transfer in an MRF manner to post-process the output of the inpainting network, creating crisp textures at the cost of heavy iterative optimization during testing. Further, Yu *et al.* [43] conducted the neural patch copy-paste operation with full convolutions, enabling texture generation in one forward pass. Instead of forcing copy-paste in the testing phase, Wang *et al.* [37] gave MRF-based non-local loss to encourage the network to model it implicitly. To better handle the generation of the missing regions, various types of intermediate representations (*e.g.* edges [27] and foreground segmentation [40]) are exploited to guide the final fine detail generation in a two-stage framework. Meanwhile, some researches focus on generating pluralistic results [48] or improving generation efficiency [31].

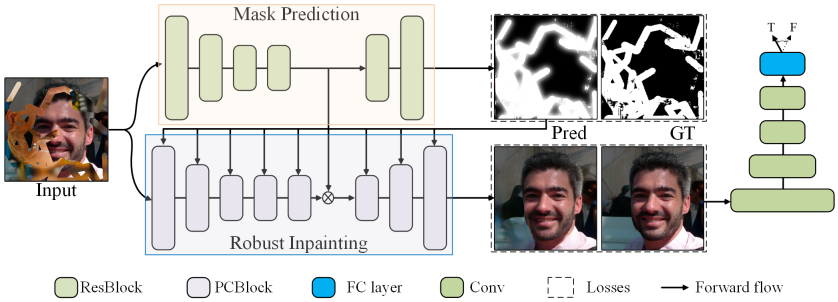


Fig. 2: Our framework. It consists of sequentially connected mask prediction and robust inpainting networks, trained in an adversarial fashion.

Also, research exists to study convolution variants [30, 33, 22, 44]. They exploit the mask more explicitly than simple concatenation with the input. Generally, the crafted networks learn upon known pixels indicated by the mask.

Other Removal Tasks A variety of removal tasks are related to blind inpainting, *e.g.* raindrop removal [29]. Their assumptions are similar regarding the condition that some pixels are clean or useful. The difference is on feature statistics of noisy areas subject to various strong priors.

3 Robust Blind Inpainting

For this task, the input is only a degraded image $\mathbf{I} \in \mathbb{R}^{h\times w\times c}$ (contaminated by unknown visual signals), and the output is expected to be a plausible image $\hat{\mathbf{O}} \in \mathbb{R}^{h\times w\times c}$, approaching ground truth $\mathbf{O} \in \mathbb{R}^{h\times w\times c}$ of $\mathbf{I}$.

The degraded image $\mathbf{I}$ in the blind inpainting setting is formulated as

$$\mathbf{I} = \mathbf{O} \odot (\mathbf{1} - \mathbf{M}) + \mathbf{N} \odot \mathbf{M}, \tag{1}$$

where $\mathbf{M} \in \mathbb{R}^{h\times w\times 1}$ is a binary region mask (with value 0 for known pixels and 1 otherwise), and $\mathbf{N} \in \mathbb{R}^{h\times w\times c}$ is a noisy visual signal. $\odot$ is the Hadamard product operator. Given $\mathbf{I}$, we predict $\hat{\mathbf{O}}$ (an estimate of $\mathbf{O}$) with latent variables $\mathbf{M}$ and $\mathbf{N}$. Also, Eq. (1) is the means to produce training tuples $< \mathbf{I}_i, \mathbf{O}_i, \mathbf{M}_i, \mathbf{N}_i >_{|i=1,\dots,m}$.

3.1 Training Data Generation

How to define the possible image contamination ($\mathbf{N}$ indicates what and $\mathbf{M}$ indicates where in Eq. (1)) is the essential prerequisite for whether a neural system could be robust to a variety of possible image contamination. Setting $\mathbf{N}$ as a constant value or certain kind of noise makes it and $\mathbf{M}$ easy to be distinguished by a deep neural net or even a simple linear classifier from a natural image patch. This prevents the model to predict inconsistent regions based on the semantic context, as drawing prediction with the statistics of a local patch should be

much easier. It converts the original blind inpainting problem into a vanilla inpainting one with a nearly perfect prediction of $\mathbf{M}$. It becomes solvable with the existing techniques. But its assumption generally does not hold in the real-world scenarios, *e.g.*, graffiti removal shown in Fig. 1.

In this regard, the key for defining $\mathbf{N}$ is to make it indistinguishable as much as possible from $\mathbf{I}$ on image pattern, so that the model cannot decide if a local patch is corrupted without seeing the image context. Then a neural system trained with such data has the potential to work on unknown contamination.

In this paper, we use real-world image patches to form $\mathbf{N}$. This ensures that local patches between $\mathbf{N}$ and $\mathbf{I}$ are indistinguishable, enforcing the model to draw an inference based on contextual information, which eventually improves the generalization ability for real-world data. Further, we alleviate any priors introduced by $\mathbf{M}$ in training via employing free-form strokes [44]. Existing blind or non-blind inpainting methods often generate the arbitrary size of a rectangle or text-shaped masks. However, this is not suitable for our task, because it may encourage the model to locate the corrupted part based on the rectangle shape. Free-form masks can largely diversify the shape of masks, making the model harder to infer corrupted regions with shape information.

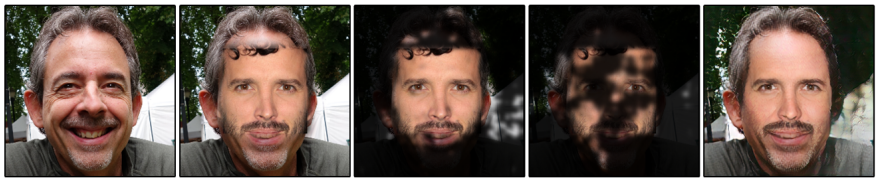
Also, we note that direct blending image $\mathbf{O}$ and $\mathbf{N}$ using Eq. (1) would lead to noticeable edges, which are strong indicators to distinguish among noisy areas. This will inevitably sacrifice the semantic understanding capability of the used model. Thus, we dilate the $\mathbf{M}$ into $\tilde{\mathbf{M}}$ by the iterative Gaussian smoothing in [37] and employ alpha blending in the contact regions between $\mathbf{O}$ and $\mathbf{N}$.

3.2 Our Method

We propose an end-to-end framework, named Visual Consistent Network (VCN) (Fig. 2). VCN has two sub-modules, *i.e.* Mask Prediction Network (MPN) and Robust Inpainting Network (RIN). MPN is to predict potential visually inconsistent areas of a given image, while RIN is to inpaint inconsistent parts based on the predicted mask and the image context. Note that these two submodules are correlated. MPN provides an inconsistency mask $\hat{\mathbf{M}} \in \mathbb{R}^{h \times w \times 1}$, where $\hat{\mathbf{M}}_p \in [0, 1]$, helping RIN locate inconsistent regions. On the other hand, by leveraging local and global semantic context, RIN largely regularizes MPN, enforcing it to focus on these regions instead of simply fitting our generated data.

Our proposed VCN is robust to blind image inpainting in the given relativistic generalized setting. Its robustness is shown in two aspects. MPN of VCN can predict the regions to be repaired with decent performance even the contamination patterns are new to the trained model. More importantly, RIN of VCN synthesizes plausible and convincing visual content for the predicted missing regions, robust against mask prediction errors. Their designs are detailed below.

Mask Prediction Network (MPN) MPN aims to learn a mapping F where $F(\mathbf{I}) \rightarrow \mathbf{M}$. MPN is with an encoder-decoder structure using residual blocks [11], and takes binary cross-entropy loss between $\hat{\mathbf{M}}$ and $\mathbf{M}$ as the optimization goal. To stabilize its learning, a self-adaptive loss is introduced to balance positive- and negative-sample classification, because clear pixels outnumber the damages



(a) (b) (c) (d) (e)

Fig. 3: Visualization of learned masks with different training strategies. (a) Input ground truth. (b) Input face image whose central part is replaced by another face with the rectangle mask. (c) Estimated mask with training MPN alone. (d) Estimated mask with joint training with inpainting network. (e) Output image of VCN. The used MPN is trained with free-form stroke masks [44].

ones ($|\{p|\mathbf{M}_p = 1\}| = \rho|\{p|\mathbf{M}_p = 0\}|$ where $\rho = 0.56 \pm 0.17$). This self-adaptive loss is expressed as

$$\mathcal{L}_m(\hat{\mathbf{M}}, \mathbf{M}) = -\tau \sum_p \mathbf{M}_p \cdot \log(\hat{\mathbf{M}}_p) - (1-\tau) \sum_q (\mathbf{1} - \mathbf{M}_q) \cdot \log(\mathbf{1} - \hat{\mathbf{M}}_q), \quad (2)$$

where $p \in \{p|\mathbf{M}_p = 1\}$, $q \in \{q|\mathbf{M}_q = 0\}$, and $\tau = |\{p|\mathbf{M}_p = 0\}|/(h \times w)$.

Note that $\hat{\mathbf{M}}$ is an estimated soft mask where $0 \leq \hat{\mathbf{M}}_p \leq 1$ for $\forall p$, although we employ a binary version for $\mathbf{M}$ in Eq. (1). It means the damaged pixels are not totally abandoned in the following inpainting process. The softness of $\hat{\mathbf{M}}$ enables the differentiability of the whole network. Additionally, it lessens error accumulation caused by pixel misclassification, since pixels whose status (damaged or not) MPN are uncertain about are still utilized in the later process.

Note that the objective of MPN is to detect all corrupted regions. Thus it tends to predict large corrupted regions for an input corrupted image, which is shown in Fig. 3(c). As a result, it makes the subsequent inpainting task too difficult to achieve. To make the task more tractable, we instead propose to detect the *inconsistency* region of the image, as shown in Fig. 3(d), which is much smaller. If these regions are correctly detected, other corrupted regions can be naturally blended to the image, leading to realistic results. In the following, we show that by jointly learning MPN with RIN, the MPN eventually locates *inconsistency* regions instead of all corrupted ones.

Robust Inpainting Network (RIN) With the $\hat{\mathbf{M}}$ located by MPN, RIN corrects them and produces a realistic result $\mathbf{O}$ – that is, RIN learns a mapping G where $G(\mathbf{I}|\hat{\mathbf{M}}) \rightarrow \mathbf{O}$. Also, RIN is structured in an encoder-decoder fashion with probabilistic contextual blocks (PCB). PCB is a residual block variant armed with a new normalization (Fig. 4), incorporating spatial information with the predicted mask.

With the predicted mask $\hat{\mathbf{M}}$, repairing corrupted regions requires knowledge inference from context, and being skeptical to the mask for error propagation from the previous stage. A naive solution is to concatenate the mask with the image and feed them to a network. However, this way captures context semantics

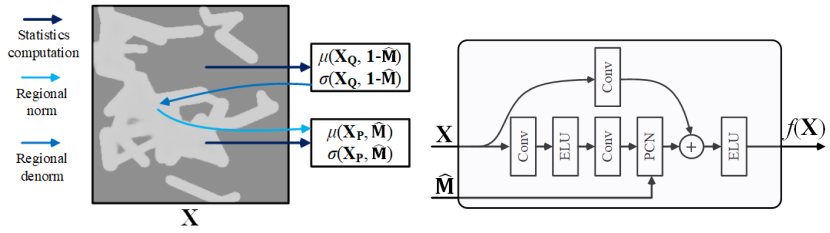


Fig. 4: Left: $\mathcal{T}(\cdot)$ in probabilistic context normalization (PCN, defined in Eq. (3)). Right: probabilistic contextual block (PCB). $\mathbf{X}$ and $\hat{\mathbf{M}}$ denote the input feature map and the predicted mask, respectively. $\mathbf{X_P} = \mathbf{X} \odot \hat{\mathbf{M}}$ and $\mathbf{X_Q} = \mathbf{X} \odot (\mathbf{I} - \hat{\mathbf{M}})$.

only in deeper layers, and does not consider the mask prediction error explicitly. To improve contextual modeling and minimize mask error propagation, it would be better if the transfer is done in all building blocks, driven by the estimated mask confidence. Hence, we propose a probabilistic context normalization (PCN, Fig. 4) to transfer contextual information in different layers.

Our PCN module is composed of the context feature transfer term and feature preserving term. The former transfers mean and variance from known features to unknown areas, both indicated by the estimated soft mask $\hat{\mathbf{M}}$ ($\mathbf{H}$ defined below is its downsampled version). It is a learnable convex combination of feature statistics from the predicted known areas and unknowns ones. Feature preserving term keeps the features in the known areas (of high confidence) intact. The formulation of PCN is given as

$$\mathrm{PCN}(\mathbf{X}, \mathbf{H}) = \underbrace{[\beta \cdot \mathcal{T}(\mathbf{X}, \mathbf{H}) \odot \mathbf{H} + (1-\beta)\mathbf{X} \odot \mathbf{H}]}_{\text{Context feature transfer}} + \underbrace{\mathbf{X} \odot \bar{\mathbf{H}}}_{\text{Feature preserving}}, \tag{3}$$

and the operator $\mathcal{T}(\cdot)$ is to conduct instance internal statistics transfer as

$$\mathcal{T}(\mathbf{X}, \mathbf{H}) = \frac{\mathbf{X_P} - \mu(\mathbf{X_P}, \mathbf{H})}{\sigma(\mathbf{X_P}, \mathbf{H})} \cdot \sigma(\mathbf{X_Q}, \bar{\mathbf{H}}) + \mu(\mathbf{X_Q}, \bar{\mathbf{H}}), \tag{4}$$

where $\mathbf{X}$ is the input feature map of PCN, and $\mathbf{H}$ is nearest-neighbor downsampled from $\hat{\mathbf{M}}$, which shares the same height and width with $\mathbf{X}$. $\bar{\mathbf{H}} = \mathbf{1} - \mathbf{H}$ indicates the regions that MPN considers clean. $\mathbf{X_P} = \mathbf{X} \odot \mathbf{H}$ and $\mathbf{X_Q} = \mathbf{X} \odot \bar{\mathbf{H}}$. β is a learnable channel-wise vector ($\beta \in \mathcal{R}^{1\times1\times c}$ and $\beta \in [0, 1]$) computed from $\mathbf{X}$ by a squeeze-and-excitation module [12] as

$$\beta = f(\bar{x}), \quad \text{and} \quad \bar{x}_k = \frac{1}{h' \times w'} \sum_{i=1}^{h'} \sum_{j=1}^{w'} \mathbf{X}_{i,j,k}, \tag{5}$$

where $\bar{x} \in \mathcal{R}^{1\times1\times c}$ is also a channel-wise vector computed by average pooling $\mathbf{X}$, and $f(\cdot)$ is the excitation function composed by two fully-connected layers with activation functions (ReLU and Sigmoid, respectively).

$\mu(\cdot,\cdot)$ and $\sigma(\cdot,\cdot)$ in Eq. (4) compute the weighted average and standard deviation respectively in the following manner:

$$\mu(\mathbf{Y},\mathbf{T}) = \frac{\sum_{i,j}(\mathbf{Y}\odot\mathbf{T})_{i,j}}{\epsilon+\sum_{i,j}\mathbf{T}_{i,j}}, \sigma(\mathbf{Y},\mathbf{T}) = \sqrt{\frac{\sum_{i,j}(\mathbf{Y}\odot\mathbf{T}-\mu(\mathbf{Y},\mathbf{T}))^2_{i,j}}{\epsilon+\sum_{i,j}\mathbf{T}_{i,j}}+\epsilon}, \quad (6)$$

where $\mathbf{Y}$ is a feature map, $\mathbf{T}$ is a soft mask with the same size of $\mathbf{Y}$, and ϵ is a small positive constant. i and j are the indexes of height and width, respectively.

Prior work [8, 15] showed that feature mean and variance from an image are related to its semantics and texture. The feature statistics propagation by PCN helps regenerate inconsistent areas by leveraging contextual mean and variance. This is intrinsically different from existing methods that implicitly achieve this goal in deep layers, as we explicitly accomplish it in each building block. Thus PCN is beneficial to the learning and performance of blind inpainting. More importantly, RIN keeps robust considering potential errors in $\hat{\mathbf{M}}$ from MPN, although RIN is guided by $\hat{\mathbf{M}}$ for repairing. This is validated in Section 4.3.

Other special design in RIN includes feature fusion and a comprehensive optimization target. Feature fusion denotes concatenating the discriminative feature (bottleneck of MPN) to the bottleneck of RIN. This not only enriches the given features to be transformed into a natural image by introducing potential spatial information, but also enhances the discriminative learning for the location problem based on the gradients from the generation procedure.

The learning objective of RIN considers pixel-wise reconstruction errors, the semantic and texture consistency, and a learnable optimization target by fooling a discriminator via generated images as

$$\mathcal{L}_g(\hat{\mathbf{O}},\mathbf{O}) = \underbrace{\lambda_r||\hat{\mathbf{O}}-\mathbf{O}||_1}_{\text{reconstruction}} + \underbrace{\lambda_s||V^l_{\hat{\mathbf{O}}}-V^l_{\mathbf{O}}||_1}_{\text{semantic consistency}} + \underbrace{\lambda_f\mathcal{L}_{mrf}(\hat{\mathbf{O}},\mathbf{O})}_{\text{texture consistency}} + \underbrace{\lambda_a\mathcal{L}_{adv}(\hat{\mathbf{O}},\mathbf{O})}_{\text{adversarial term}}, \quad (7)$$

where $\hat{\mathbf{O}} = G(\mathbf{I}|\hat{\mathbf{M}})$. V is a pre-trained VGG19 network. $V^l_{\mathbf{O}}$ means we extract the feature layer l (ReLU3_2) of the input $\mathbf{O}$ when $\mathbf{O}$ is passed into V. Besides, λ_r, λ_s, λ_f, and λ_a are regularization coefficients to adjust each term influence, and they are set to 1.4, $1e-4$, $1e-3$, and $1e-3$ in our experiments, respectively.

ID-MRF loss [37, 25] is employed as our texture consistency term. It computes the sum of the patch-wise difference between neural patches from the generated content and those from the corresponding ground truth using a relative similarity measure. It enhances generated image details by minimizing discrepancy with its most similar patch from the ground truth.

For the adversarial term, WGAN-GP [10, 1] is adopted as

$$\mathcal{L}_{adv}(\hat{\mathbf{O}},\mathbf{O}) = -E_{\hat{\mathbf{O}}\sim\mathbb{P}_{\hat{\mathbf{O}}}}[D(\hat{\mathbf{O}})], \quad (8)$$

where $\mathbb{P}$ denotes data distribution, and D is a discriminator for the adversarial training. Its corresponding learning objective for the discriminator is given as

$$\mathcal{L}_D(\hat{\mathbf{O}},\mathbf{O}) = E_{\hat{\mathbf{O}}\sim\mathbb{P}_{\hat{\mathbf{O}}}}[D(\hat{\mathbf{O}})] - E_{\mathbf{O}\sim\mathbb{P}_{\mathbf{O}}}[D(\mathbf{O})] + \lambda_{gp}E_{\tilde{\mathbf{O}}\sim\mathbb{P}_{\tilde{\mathbf{O}}}}[(||\nabla_{\tilde{\mathbf{O}}}D(\tilde{\mathbf{O}})||_2-1)^2], \quad (9)$$

where $\tilde{\mathbf{O}} = t\hat{\mathbf{O}}+(1-t)\mathbf{O}$, $t\in[0,1]$, and $\lambda_{gp}=10$.

3.3 Training Procedure

Generation of training data is given in Eq. (1), where production of $\mathbf{M}$ is adopted from [44] as free-form strokes. The final prediction of our model is $G(\mathbf{I}|F(\mathbf{I}))$. All input and output are linearly scaled within range $[-1, 1]$.

There are two training stages. MPN and RIN are separately trained at first. After both networks are converged, we jointly optimize $\min_{\theta_F,\theta_G} \lambda_m \mathcal{L}_m(F(\mathbf{I}), \mathbf{M}) + \mathcal{L}_g(G(\mathbf{I}|F(\mathbf{I})), \mathbf{O})$ with $\lambda_m = 2.0$.

4 Experimental Results and Analysis

Our model and baselines are implemented using Tensorflow (v1.10.1). The evaluation platform is a Linux server with an Intel Xeon E5 (2.60GHz) CPU and an NVidia TITAN X GPU. Our full model (MPN + RIN) has 3.79M parameters and costs around 41.64ms to process a 256×256-size RGB image.

The datasets include FFHQ (faces) [17], CelebA-HQ (faces) [16], ImageNet (objects) [6], and Places2 (scenes) [49]. Our training images are all with size 256×256 unless otherwise specified. For FFHQ, images are downsampled from the original 1024×1024. For ImageNet and Places2, central cropping and padding are applied. When training on FFHQ, its corresponding noisy images are drawn from the training sets of CelebA-HQ and ImageNet. For training on ImageNet and Places2, these two datasets are the noisy source for each other.

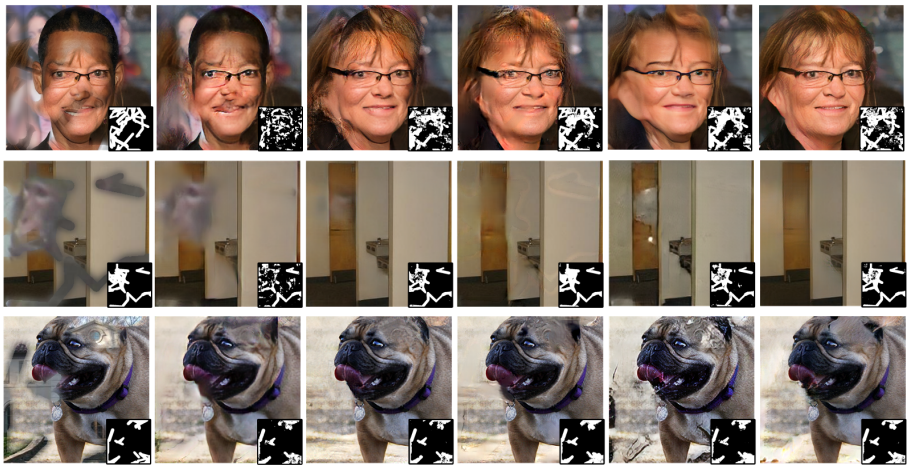
Our baselines are all based on GAN [9] frameworks. We construct four alternative models to show the influence brought by the network architecture and module design. For a fair comparison, they are all equipped with mask prediction network (MPN) in front of their input, and are trained from scratch (MPNs are trained in the same way explained in Section 3.3). The first two alternatives are built upon the contextual attention (CA) model [43] and generative multi-column (GMC) model [37]. The input of these two inpainting variants is the concatenation of the estimated soft mask and the noisy image. The last two baselines are by employing the partial convolution (PC) [22] and gated convolution (GC) [44] as their basic building blocks, respectively, to construct the network, intending to explore how the used neural unit affects this blind inpainting. Compared with our VCN (3.79M), the model complexity of these CA, GMC, PC, and GC baselines is high as 4.86M, 13.7M, 4.69M, and 6.06M, respectively. All these numbers already include the model complexity of MPN (1.96M).

4.1 Mask Estimation Evaluation

We evaluate the mask prediction performance of all used methods based on their computed binary cross-entropy (BCE) loss (the lower the better) on the testing sets. As shown in Table 1, our VCN achieves superior performance compared to GC [44], PC [22], GMC [37], and CA [43], except that our SSIM in ImageNet-4K is slightly lower than GMC. It shows that different generative structures and modules affect not only generation but also the relevant mask estimation

Table 1: Quantitative results on the testing sets from different methods.

Method	FFHQ-2K			Places2-4K			ImageNet-4K		
	BCE↓	PSNR↑	SSIM↑	BCE↓	PSNR↑	SSIM↑	BCE↓	PSNR↑	SSIM↑
CA [43]	1.297	16.56	0.5509	0.574	18.12	0.6018	0.450	17.68	0.5285
GMC [37]	0.766	20.06	0.6675	0.312	20.38	0.6956	0.312	19.56	**0.6467**
PC [22]	**0.400**	20.19	0.6795	0.273	19.73	0.6682	0.229	19.53	0.6277
GC [44]	0.660	17.16	0.5915	0.504	18.42	0.6423	0.410	18.35	0.6416
Our VCN	**0.400**	**20.94**	**0.6999**	**0.253**	**20.54**	**0.6988**	**0.226**	**19.58**	0.6339



(a) Input image (b) CA [43] (c) GMC [37] (d) PC [22] (e) GC [44] (f) Our results

Fig. 5: Visual comparison on synthetic data from FFHQ (top), Places2 (middle), and ImageNet (bottom). The ground truth masks (shown in the first column) and the estimated ones (in binary form) are shown on the bottom right corner of each image.

performance. Clearly, VCN with spatial normalization works decently, benefiting mask prediction by propagating clean pixels to the damaged areas.

Partial convolution (in PC [22]) yields relatively lower performance, and direct concatenation between the estimated mask and the input image (used in CA [43] and GMC [37]) is least effective. Visual comparison of the predicted masks of different methods is included in Fig. 5, where the results from PC and VCN are comparable. They look better than those of CA, GMC, and GC.

4.2 Blind Inpainting Evaluation

Synthetic Experiments Visual comparison of the used baselines and our method on the synthetic data (composed in the way we describe in Sec. 3.1) are given in Fig. 5. Our method produces more visually convincing results with fewer artifacts, which are not much disturbed by the unknown contamination areas. On the other hand, the noisy areas from CA and GMC baselines manifest

(a) Input (b) Our result (a) Input (b) Our result

Fig. 6: Our results on facades and faces with other shaped masks.

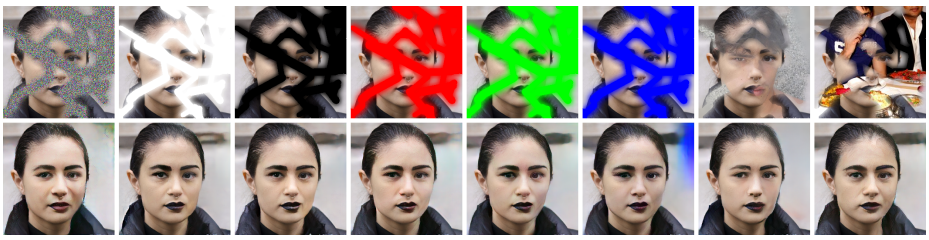


Fig. 7: Visual results on FFHQ with masks filled with different contents. First row: input; second row: corresponding results from our model. Last two images are filled with content drawn from the testing sets of CelebA-HQ and ImageNet respectively.

that concatenation of mask and input to learn the ground truth is not an effective way for blind inpainting setting. More cases are given in the supplementary file.

About randomly inserted patches or text shape masks, Fig. 6 shows that our method can locate the inserted car, complete the facade (train/test on Paris Streetview [28]), and restore text-shape corrupted regions on the testing image from FFHQ.

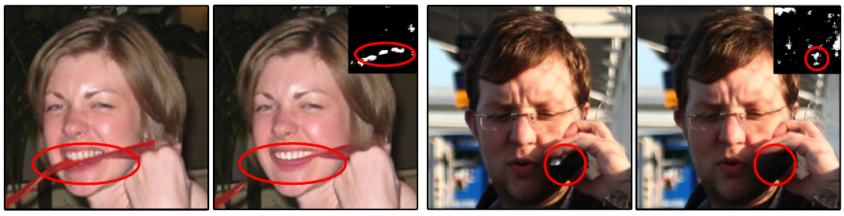
Robustness against Various Degradation Patterns Our training scheme makes the proposed model robust to fill content, as shown in Fig. 7. It can deal with Gaussian noise or constant color filling directly, while these patterns are not included in our training. This also shows such a training scheme makes the model learn to tell and inpaint visual inconsistency instead of memorizing synthetic missing data distribution.

PSNR and SSIM index evaluated on the testing sets of the used datasets are given in Table 1 for reference. Generally, VCN yields better or comparable results compared with baselines, verifying the effectiveness of spatial normalization about image fidelity restoration in this setting.

Further, pairwise A/B tests are adopted for blind user studies using Google Forms. 50 participants are invited for evaluating 12 questionnaires. Each has 40 pairwise comparisons, and every comparison shows results predicted from two different methods based on the same input, randomized in the left-right order. As given in Table 2, our method outperforms the CA, PC, and GC in all datasets

Table 2: User studies. Each entry gives the percentage of cases where results by our approach are judged as more realistic than another solution. The observation and decision time for users is unlimited.

Methods	VCN > CA	VCN > GMC	VCN > PC	VCN > GC
FFHQ	99.64%	80.83%	77.66%	92.15%
Places2	81.39%	51.63%	70.49%	78.15%
ImageNet	91.20%	50.09%	77.92%	83.30%



(a) Input (b) Our result (a) Input (b) Our result

Fig. 8: Blind inpainting on the real occluded faces from COCO dataset with VCN. Red ellipses in the pictures highlight the regions to be edited.

Fig. 9: Visual evaluation on raindrop removal dataset. Left: Input image. Middle: AttentiveGAN [29]. Right: Ours (Best view in original resolution).

and GMC in FFHQ, and yields comparable visual performance with GMC on ImageNet and Places2 with a much smaller model size (3.79M vs. 13.70M).

Blind Inpainting on Real Cases Fig. 8 gives blind inpainting (trained on FFHQ) on the occluded face from COCO dataset [21]. Note VCN can automatically, and at least partially, restore these detected occlusions. The incomplete removal with red strip bit in the mouth may be caused by similar patterns in FFHQ, as mentioned that the detected visual inconsistency is inferred upon the learned distribution from the training data.

Model Generalization We evaluate the generality of our model on raindrop removal with a few training data. The dataset in [29] gives paired data (noisy and clean ones) without masks. Our full model (pre-trained on Places2 with random strokes) achieves promising qualitative results (Fig. 9) on the testing set, which is trained with a few training images (20 RGB images of resolution 480×720, around 2.5% training data). In the same training setting, testing results by AttentiveGAN [29] (a raindrop removal method) yield 24.99dB while ours is

Table 3: Quantitative results of component ablation of VCN on FFHQ (ED: Encoder-decoder; fusion: the bottleneck connection between MPN and RIN; -RM: removing the estimated contamination as $G(\mathbf{I} \odot (\mathbf{1} - \hat{\mathbf{M}})|\hat{\mathbf{M}})$; SC: semantic consistency term).

Model	ED	VCN w/o MPN	VCN w/o fusion	VCN w/o SC	VCN-RM	VCN full
PSNR↑	19.43	18.88	20.06	20.56	20.87	**20.94**
SSIM↑	0.6135	0.6222	0.6605	0.6836	**0.7045**	0.6999
BCE↓	-	-	0.560	0.653	0.462	**0.400**

(a) (b) (c) (d) (e) (f)

Fig. 10: Visual comparison on FFHQ using VCN variants. (a) Input image. (b) VCN w/o MPN. (c) VCN w/o skip. (d) VCN w/o semantics. (e) VCN-RM. (f) VCN full.

Fig. 11: Visual editing (face-swap) on FFHQ. First row: image with coarse editing where a new face (from CelebA-HQ) is pasted at the image center; Second row: corresponding results from our model. Best viewed with zoom-in.

26.22dB. It proves the learned visual consistency ability can be transferred to other similar removal tasks with a few target data.

4.3 Ablation Studies

w and w/o MPN Without MPN, fidelity restoration of VCN degrades a lot in Table 3. The comparison in Fig. 10(b) shows VCN w/o MPN finds obvious artifacts like blue regions. But it fails to completely remove the external introduced woman face. Thus our introduced task decomposition and joint training strategy are effective. Compared with the performance of ED, VCN variants show the superiority of the module design in our solution.

Fusion of Discriminative and Generative Bottlenecks Improvement of such modification on mask prediction (BCE), PSNR, and SSIM is limited. But this visual improvement shown in Fig. 10(c) and (f) is notable. Such a shortcut significantly enhances detail generation.

Input for Inpainting Since the filling mask is estimated instead of given, removing possible contamination areas may degrade the generation performance due to the mask prediction error. Fig. 10(e) validates our consideration.
Loss Discussion Significance of the semantic consistency term that affects VCN is given in Table 10. It shows that this term benefits the discrimination ability and fidelity restoration of the model since removing it leads to a decrease of PSNR (from 20.94 to 20.56) and SSIM (from 0.6999 to 0.6836), and increase of BCE (from 0.4 to 0.653). Removing this term leads to hair and texture artifacts as shown in Fig. 10(d). Other terms have been discussed in [43, 37].
Study of PCN In the testing phase, we adjust ρ (instead of using the trained one) manually in PCN to show its flexibility in controlling interference of possible contamination (in the supplementary file). With increasing ρ, VCN tends to generate missing parts based on context instead of blending the introduced 'noise'.
Applications on Image Blending Our blind inpainting system also finds applications on image editing, especially on blending user-fed visual material with the given image automatically. Our method can utilize the filling content to edit the original ones. The given material from external datasets is adjusted on its shape, color, shadow, and even its semantics to appeal to its new context, as given in Fig. 11. The editing results are natural and intriguing. Note the estimated masks mainly highlight the outlines of the pasted rectangle areas, which are just inconsistent regions according to context.
Limitation and Failure Cases If contaminated areas in images are large enough to compromise main semantics, our model cannot decide which part is dominant and the performance would degrade dramatically. Some failure cases are given in the supplementary file. Moreover, if users want to remove a certain object from an image, it would be better to feed the users mask into the robust inpainting network to complete the target regions. On the other hand, our method cannot repair the common occlusion problems (like human body occlusion) because our model does not regard this as an inconsistency.

5 Conclusion

We have proposed a robust blind inpainting framework with promising restoration ability on several benchmark datasets. We designed a new way of data preparation, which relaxes missing data assumptions, as well as an end-to-end model for joint mask prediction and inpainting. A novel probabilistic context normalization is used for better context learning. Our model can detect incompatible visual signals and transform them into contextual consistent ones. It is suitable to automatically repair images when manually labeling masks is hard. Our future work will be to explore the transition between common inpainting and blind inpainting, *e.g.* using coarse masks or weakly supervised hints to guide the process.

References

1. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: ICML. pp. 214–223 (2017)
2. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: Patchmatch: A randomized correspondence algorithm for structural image editing. TOG **28**(3), 24 (2009)
3. Cai, N., Su, Z., Lin, Z., Wang, H., Yang, Z., Ling, B.W.K.: Blind inpainting using the fully convolutional neural network. The Visual Computer **33**(2), 249–261 (2017)
4. Criminisi, A., Pérez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. TIP **13**(9), 1200–1212 (2004)
5. Darabi, S., Shechtman, E., Barnes, C., Goldman, D.B., Sen, P.: Image melding: Combining inconsistent images using patch-based synthesis. TOG **31**(4), 82 (2012)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. pp. 248–255 (2009)
7. Dong, B., Ji, H., Li, J., Shen, Z., Xu, Y.: Wavelet frame based blind image inpainting. Applied and Computational Harmonic Analysis **32**(2), 268–279 (2012)
8. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: CVPR. pp. 2414–2423 (2016)
9. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NeurIPS. pp. 2672–2680 (2014)
10. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. In: NeurIPS. pp. 5769–5779 (2017)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
12. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: CVPR. pp. 7132–7141 (2018)
13. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Globally and locally consistent image completion. TOG **36**(4), 107 (2017)
14. Jia, J., Tang, C.K.: Image repairing: Robust image synthesis by adaptive nd tensor voting. In: CVPR (2003)
15. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV. pp. 694–711 (2016)
16. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196 (2017)
17. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. arXiv preprint arXiv:1812.04948 (2018)
18. Kopf, J., Kienzle, W., Drucker, S., Kang, S.B.: Quality prediction for image completion. TOG **31**(6), 131 (2012)
19. Levin, A., Zomet, A., Weiss, Y.: Learning how to inpaint from global image statistics. In: ICCV (2003)
20. Li, Y., Liu, S., Yang, J., Yang, M.H.: Generative face completion. In: CVPR. pp. 3911–3919 (2017)
21. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
22. Liu, G., Reda, F.A., Shih, K.J., Wang, T.C., Tao, A., Catanzaro, B.: Image inpainting for irregular holes using partial convolutions. In: ECCV. pp. 85–100 (2018)

23. Liu, H., Jiang, B., Xiao, Y., Yang, C.: Coherent semantic attention for image inpainting. In: ICCV. pp. 4170–4179 (2019)
24. Liu, Y., Pan, J., Su, Z.: Deep blind image inpainting. arXiv preprint arXiv:1712.09078 (2017)
25. Mechrez, R., Talmi, I., Zelnik-Manor, L.: The contextual loss for image transformation with non-aligned data. arXiv preprint arXiv:1803.02077 (2018)
26. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. arXiv preprint arXiv:1802.05957 (2018)
27. Nazeri, K., Ng, E., Joseph, T., Qureshi, F., Ebrahimi, M.: Edgeconnect: Generative image inpainting with adversarial edge learning. arXiv preprint arXiv:1901.00212 (2019)
28. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: CVPR. pp. 2536–2544 (2016)
29. Qian, R., Tan, R.T., Yang, W., Su, J., Liu, J.: Attentive generative adversarial network for raindrop removal from a single image. In: CVPR. pp. 2482–2491 (2018)
30. Ren, J.S., Xu, L., Yan, Q., Sun, W.: Shepard convolutional neural networks. In: NeurIPS. pp. 901–909 (2015)
31. Sagong, M.c., Shin, Y.g., Kim, S.w., Park, S., Ko, S.j.: Pepsi: Fast image inpainting with parallel decoding network. In: CVPR. pp. 11360–11368 (2019)
32. Sun, J., Yuan, L., Jia, J., Shum, H.Y.: Image completion with structure propagation. In: TOG. vol. 24, pp. 861–868 (2005)
33. Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., Geiger, A.: Sparsity invariant cnns. In: 3DV. pp. 11–20 (2017)
34. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: CVPR. pp. 8798–8807 (2018)
35. Wang, Y., Chen, Y.C., Zhang, X., Sun, J., Jia, J.: Attentive normalization for conditional image generation. In: CVPR. pp. 5094–5103 (2020)
36. Wang, Y., Szlam, A., Lerman, G.: Robust locally linear analysis with applications to image denoising and blind inpainting. SIAM Journal on Imaging Sciences **6**(1), 526–562 (2013)
37. Wang, Y., Tao, X., Qi, X., Shen, X., Jia, J.: Image inpainting via generative multi-column convolutional neural networks. In: NeurIPS (2018)
38. Wang, Y., Tao, X., Shen, X., Jia, J.: Wide-context semantic image extrapolation. In: CVPR. pp. 1399–1408 (2019)
39. Xie, C., Liu, S., Li, C., Cheng, M.M., Zuo, W., Liu, X., Wen, S., Ding, E.: Image inpainting with learnable bidirectional attention maps. arXiv preprint arXiv:1909.00968 (2019)
40. Xiong, W., Yu, J., Lin, Z., Yang, J., Lu, X., Barnes, C., Luo, J.: Foreground-aware image inpainting. In: CVPR. pp. 5840–5848 (2019)
41. Yan, M.: Restoration of images corrupted by impulse noise and mixed gaussian impulse noise using blind inpainting. SIAM Journal on Imaging Sciences **6**(3), 1227–1245 (2013)
42. Yang, C., Lu, X., Lin, Z., Shechtman, E., Wang, O., Li, H.: High-resolution image inpainting using multi-scale neural patch synthesis. In: CVPR. p. 3 (2017)
43. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Generative image inpainting with contextual attention. arXiv preprint arXiv:1801.07892 (2018)
44. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Free-form image inpainting with gated convolution. In: ICCV. pp. 4471–4480 (2019)
45. Zeng, Y., Fu, J., Chao, H., Guo, B.: Learning pyramid-context encoder network for high-quality image inpainting. In: CVPR. pp. 1486–1494 (2019)

46. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. arXiv preprint arXiv:1805.08318 (2018)
47. Zhang, S., He, R., Sun, Z., Tan, T.: Demeshnet: Blind face inpainting for deep meshface verification. IEEE Transactions on Information Forensics and Security **13**(3), 637–647 (2017)
48. Zheng, C., Cham, T.J., Cai, J.: Pluralistic image completion. In: CVPR. pp. 1438–1447 (2019)
49. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. TPAMI (2017)