# DECONTAMINATION TRANSFORMER FOR BLIND IMAGE INPAINTING

*Chun-Yi Li* *Yen-Yu Lin* *Wei-Chen Chiu*

Department of Computer Science, National Chiao Tung University, Taiwan

## ABSTRACT

Blind image inpainting aims at recovering the content from a corrupted image in which the mask indicating the corrupted regions is not available in inference time. Inspired that most existing methods for inpainting suffer from complex contamination, we propose a model that explicitly predicts the real-valued alpha mask and contaminant to eliminate the contamination from the corrupted image, thus improving the inpainting performance. To enhance the overall semantic consistency, the attention mechanism of transformers is exploited and integrated into our inpainting network. We conduct extensive experiments to verify our method against blind and non-blind inpainting models and demonstrate its effectiveness and generalizability to different sources of contaminant.

***Index Terms***— Blind image inpainting, Transformer

## 1. INTRODUCTION

Image inpainting as the task of recovering the missing regions in a corrupted image has wide applications such as image restoration and editing. While there exists numerous prior works [3, 15, 13, 7, 9, 16, 20, 17, 6] being capable of filling the missing regions by predicting the image content consistent with the context (i.e. uncorrupted regions), they generally require the user to provide the mask indicating the missing regions, and such a mask is typically restricted to be binary (i.e. pixels are either corrupted or uncorrupted). The requirement of manually labeling the mask is time-consuming to fulfill, and the mask in binary form actually disregards the real-world cases of having some pixels half-corrupted (e.g. the contamination on the original image could be translucent) thus making their settings less practical.

To tackle the aforementioned issues for the classical image inpainting problem, the task of **blind image inpainting** emerges where the mask indicating missing regions is not required during inference, which implies that an inpainting model needs to identify where to paint as well as what to paint. While the early works of this task [1, 8] have the oversimplified assumption that the corrupted regions are filled with constant color (mostly black) or Gaussian noise thus being unrealistic, VCNet [12], as a seminal work and state-of-the-art method for blind image inpainting, recommends a more practical but challenging problem scenario together with proposing an effective model to tackle it. The new scenario assumes that the image that we would like to recover is corrupted by contaminants. Such contaminants are generated by random strokes filled with natural textures thus being more complicated than the simple black or noisy pixels.

According to this problem scenario, VCNet proposes a generative procedure of synthesizing data, where the contaminant is sampled from real-world images followed by generating the corrupted image via alpha blending between the contaminant and original image, in which the alpha mask is built by iterative Gaussian smoothing on a binary free-form mask. VCNet explicitly predicts the mask and performs inpainting via two subnetworks in a stage-wise manner, where a discriminative subnetwork takes the corrupted image to estimate the area of contamination by pixel-wise binary classification, and the inpainting subnetwork treats the estimated mask as a condition to guide inpainting. The inpainting subnetwork follows a convolutional autoencoder structure and the mask condition is injected into the subnetwork layers via spatial normalization blocks.

Although VCNet [12] shows reasonable performance on blind image inpainting, there are still several issues in its design. In this work, we particularly advance to tackle these issues for achieving better and more robust inpainting results: 1) In previous methods, e.g. VCNet and [11], the predicted mask is typically binary to separate the contamination from the uncorrupted regions, in which the pixels blended between the contaminant and the original images are not well handled thus the information related to the original image in these pixels are regrettably wasted. By contrast, the mask prediction network (MPN) in our proposed method outputs the real-valued mask; 2) In addition to discovering the blended pixels, our method differs from prior works [12, 11, 19] to have a contaminant prediction network (CPN) for estimating the content/appearance of contaminant, in which knowing what (i.e. the contaminant) is blended with the original image and how do they blend (i.e. the real-valued mask) would significantly facilitate inpainting the original image; 3) While previous methods mostly use convolutional neural networks to perform inpainting where the limited receptive fields in individual layers make it difficult to maintain long-range semantic consistency between the inpainted and uncorrupted regions, we integrate the transformer-based architecture into

Our source code and more details can be found in `https://lcy0307.github.io/Decontamination_Transformer`.

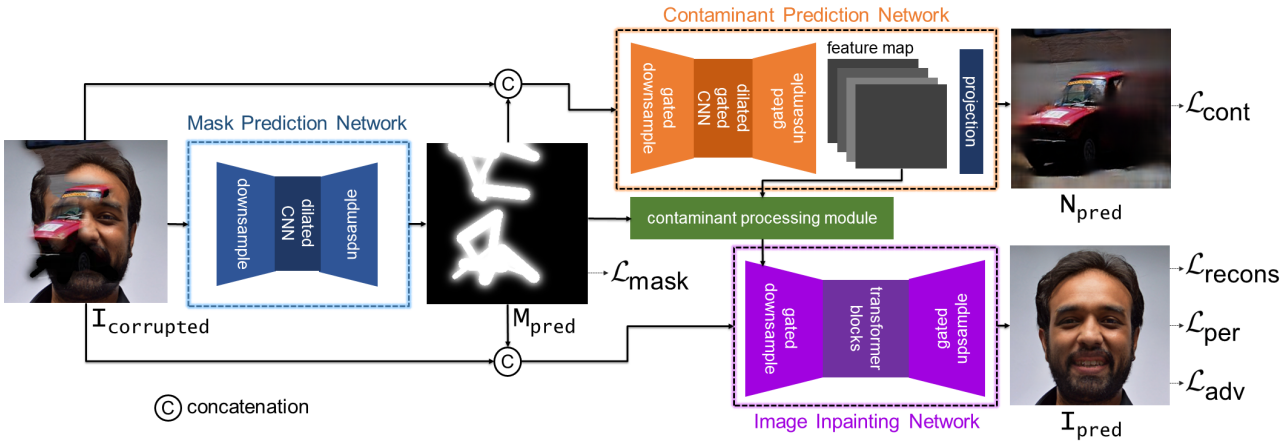


**Fig. 1**. **Our De-Contamination Model** consists of three main network components: mask prediction network (MPN), contaminant prediction network (CPN), and image inpainting network (IIN). Please refer to Sec. 2 for the detailed description.

our image inpainting network (IIN) for explicitly considering the global relationships among image regions, where the features of the estimated contaminant and predicted mask are utilized to realize inpainting. Extensive experimental results show that our model outperforms the state-of-the-arts.

## 2. METHOD

**Training Data Generation.** As motivated previously, we aim to tackle the same practical but challenging problem scenario proposed by VCNet [12] for blind image inpainting, and we follow their procedure similarly for training data generation:

$$\mathbf{I}_{\text{corrupted}} = \mathbf{I}_{\text{gt}} \odot (1 - \mathbf{M}_{\text{g}}) + \mathbf{N} \odot \mathbf{M}_{\text{g}}, \tag{1}$$

where a corrupted image $\mathbf{I}_{\text{corrupted}}$ is synthesized by alpha blending between the original image $\mathbf{I}_{\text{gt}}$ and a contaminant $\mathbf{N}$ (e.g. another real-world image). $\odot$ denotes the Hadmard product operator, and the alpha mask $\mathbf{M}_{\text{g}}$ is made by iterative Gaussian smoothing on a binary mask of random strokes (with value 1 and 0 indicating corrupted and uncorrupted regions respectively). In this paper, we adopt numerous Gaussian kernels for the smoothing operations during building masks in order to increase the variance of training data.

**De-Contamination Model.** As shown in Fig. 1, our proposed method for blind image inpainting is composed of three components: ***mask prediction network*** (MPN), ***contaminant prediction network*** (CPN), and ***image inpainting network*** (IIN). These components are detailed as follows.

Our mask prediction network (MPN) takes the corrupted image $\mathbf{I}_{\text{corrupted}}$ as input and predicts the alpha mask $\mathbf{M}_{\text{pred}}$, in which we drive its training by minimizing its errors with respect to the ground-truth mask $\mathbf{M}_{\text{g}}$ with the objective:

$$\mathcal{L}_{\text{mask}} = \|\mathbf{M}_{\text{pred}} - \mathbf{M}_{\text{g}}\|_2. \tag{2}$$

Note that our predicted mask is with real-valued pixels (i.e. $\mathbf{M}_{\text{pred}}(i,j) \in [0,1]$) to particularly take the alpha blending weights between the original image and contaminant into consideration, while VCNet [12] adopts binary pixel-wise classification for mask prediction thus neglecting the rich information hidden behind the blended pixels. The architecture of our MPN is an encoder-decoder structure where both encoder and decoder are built by residual CNN blocks with the bottleneck composed of dilated convolutions to increase the receptive fields, noting that our MPN is similar to the one in [12] but additionally equipped with batch normalization at each residual block for better training stability.

As the blended pixels (with real-valued blending weights $\in [0,1]$) in the corrupted image $\mathbf{I}_{\text{corrupted}}$ contain the appearance from contaminant $\mathbf{N}$, and the semantic inconsistency between the contaminant and original image is the key to separate them for further recovering the original image (especially on the regions of blended pixels, i.e. the boundary between corrupted and uncorrupted regions), reconstructing the content/appearance of the contaminant is thus considered helpful for inpainting the corrupted input. Our contaminant prediction network (CPN) realizes such an idea by taking the corrupted image $\mathbf{I}_{\text{corrupted}}$ and the alpha mask $\mathbf{M}_{\text{pred}}$ predicted by MPN as the input to reconstruct the contaminant $\mathbf{N}$. The architecture of CPN is similar to MPN but it particularly adopts gated convolutions [16] on the encoder and the decoder to focus on the corrupted regions (including blended pixels) with the guidance of $\mathbf{M}_{\text{pred}}$. Moreover, the skip connections are used across the encoder and the decoder to better preserve the image details.

As the main focus of CPN is to estimate the appearance of contaminant $\mathbf{N}$ on the corrupted regions, a binary contamination mask $\mathbf{M}_{\text{Con}}$ is built with value 1 indicating the partially or fully contaminated pixels and 0 for uncorrupted ones, the objective function for training CPN is then defined by

$$\mathcal{L}_{\text{cont}} = \|\mathbf{M}_{\text{Con}} \odot (\mathbf{N}_{\text{pred}} - \mathbf{N})\|_1, \tag{3}$$

where $\mathbf{N}_{\text{pred}}$ denotes the contaminant predicted by our CPN.

After having both $\mathbf{N}_{\text{pred}}$ and $\mathbf{M}_{\text{pred}}$ estimated (i.e. what is blended with and how it is blended with the original image, respectively), our image inpainting network (IIN) leverages them for inpainting on the corrupted input $\mathbf{I}_{\text{corrupted}}$ to produce the final inpainting result $\mathbf{I}_{\text{pred}}$. While VCNet only predicts the binary blending mask and does not estimate the contaminant, and its inpainting network is overloaded to simultaneously remove the contamination and perform inpainting on corrupted regions, our proposed model can firstly adopt $\mathbf{N}_{\text{pred}}$ (thanks to our CPN) and $\mathbf{M}_{\text{pred}}$ to erase the contaminant from $\mathbf{I}_{\text{corrupted}}$. Hence, our IIN is able to concentrate on filling the missing regions (i.e. where inpainting should apply) and finally achieves better inpainting performance.

Erasing contaminant from $\mathbf{I}_{\text{corrupted}}$ is realized with the help of our ***contaminant processing module*** (cf. the green rectangle in Fig. 1). The feature map of contaminant $\mathbf{N}_{\text{pred}}$ is weighted by $\mathbf{M}_{\text{pred}}$ as the conditional input for IIN (noting that the combination of $\mathbf{N}_{\text{pred}}$ and $\mathbf{M}_{\text{pred}}$ by this module happens in the feature space and we adopt a few convolution blocks to align their dimensions), subtracting such conditional input from the the features of $\mathbf{I}_{\text{corrupted}}$ extracted by the first block of IIN's encoder then achieves the contaminant erasing, where the following network blocks of IIN take over to perform the further inpainting operations.

Our IIN is also an encoder-decoder structure with several important model designs: 1) As the input to IIN is the concatenation of $\mathbf{I}_{\text{corrupted}}$ and $\mathbf{M}_{\text{pred}}$ (where the real-valued $\mathbf{M}_{\text{pred}}$ provides guidance on "where" and "how much" to inpaint), the encoder and decoder are constructed by gated convolution blocks. Moreover, the skip connections are used across the symmetric blocks between the encoder and the decoder; 2) For maximally maintaining the overall content consistency of $\mathbf{I}_{\text{pred}}$ (particularly among inpainted and uncorrupted regions) and tackling the restricted receptive field of convolutions, we propose to adopt transformer blocks in the bottleneck of IIN between the encoder and decoder. Specifically, to handle the common issues of being large computational costly and memory demanding for typical transformers, we are inspired by the fast and efficient hybrid architecture proposed in Stripformer [10] and use the intra/inter attention blocks for building our transformer module.

The inpainting output $\mathbf{I}_{\text{pred}}$ is trained to be as close to the ground truth $\mathbf{I}_{\text{gt}}$ as possible, where three objectives from different perspectives are adopted to achieve so: 1) L1 reconstruction loss $\mathcal{L}_{\text{recons}} = \|\mathbf{I}_{\text{pred}} - \mathbf{I}_{\text{gt}}\|_1$ in the pixel space, 2) perceptual loss $\mathcal{L}_{\text{per}}$ in the pretrained VGG-16 feature space, and 3) local adversarial loss $\mathcal{L}_{\text{adv}}$ between distributions of $\mathbf{I}_{\text{pred}} \odot \mathbf{M}_{\text{g}} + \mathbf{I}_{\text{gt}} \odot (1 - \mathbf{M}_{\text{g}})$ and $\mathbf{I}_{\text{gt}}$ (noting that we skip the detailed description on $\mathcal{L}_{\text{per}}$ and $\mathcal{L}_{\text{adv}}$ due to the limited space, as their formulations are identical to those in [12], where the only slight modification is to use non-saturating loss for $\mathcal{L}_{\text{adv}}$ instead of WGAN-GP for more stable training).

We will release our source code, learnt models, and all the experimental/implementation details once paper acceptance.

| Dataset | Method | PSNR↑ | SSIM↑ | LPIPS↓ |
|---|---|---|---|---|
| FFHQ | PEN-Net + $\mathbf{M}_{\text{g}}$ | 21.17 | 0.6852 | 0.2486 |
| | RFR + $\mathbf{M}_{\text{g}}$ | 23.21 | 0.8209 | 0.1087 |
| | VCNet | 21.28 | 0.7544 | 0.1706 |
| | Our proposed model | **23.70** | **0.8417** | **0.0889** |
| Places2 | PEN-Net + $\mathbf{M}_{\text{g}}$ | 19.11 | 0.5740 | 0.4546 |
| | RFR + $\mathbf{M}_{\text{g}}$ | 21.69 | 0.7720 | 0.2190 |
| | VCNet | 20.29 | 0.7204 | 0.2244 |
| | Our proposed model | **22.74** | **0.7999** | **0.1742** |
| ImageNet | PEN-Net + $\mathbf{M}_{\text{g}}$ | 18.77 | 0.5224 | 0.5124 |
| | RFR + $\mathbf{M}_{\text{g}}$ | 20.87 | 0.7442 | 0.2210 |
| | VCNet | 19.58 | 0.6690 | 0.2287 |
| | Our proposed model | **22.23** | **0.7820** | **0.1713** |

**Table 1**. Quantitative comparison among our method, state-of-the-art blind inpainting method VCnet [12], and two representative non-blind inpainting approaches (i.e. RFR [6] and PEN-Net [17], where the groundtruth alpha blending mask $\mathbf{M}_{\text{g}}$ are provided for their training and testing).

## 3. EXPERIMENTS

**Datasets & Metrics.** Experiments are conducted on the FFHQ [5], CelebA-HQ [4], Places2 [21], and ImageNet [2] datasets. All training images are of size 256×256, where images in FFHQ (Places2/ImageNet, respectively) are resized (center-cropped and zero-padded, respectively) to fit such a requirement. While experimenting on FFHQ with 68000 training and 1000 testing images, the contaminant is sampled from the CelebA-HQ and ImageNet datasets. For experimenting on the Places2 and ImageNet datasets (both with 1000 testing images), they mutually server as the source of drawing contaminant samples for each other. For evaluation, PSNR, SSIM [14], and LPIPS [18] are adopted to measure the qualities of the blind image inpainting results.

**Quantitative Evaluation.** We compare our proposed model with the state-of-the-art blind image inpainting method VCnet [12] as well as two representative non-blind inpainting methods (i.e. RFR [6] and PEN-Net [17]) with the groundtruth alpha blending mask $\mathbf{M}_{\text{g}}$ being provided in their training and testing. The quantitative results are summarized in Table 1. It can be clearly seen that our proposed method not only outperforms VCNet by a significant margin across all the datasets but also consistently achieves superior results in comparison to the non-blind inpainting baselines (i.e. RFR and PEN-Net) even when they are provided with the groundtruth alpha masks, thus verifying the efficacy of our proposed method for blind image inpainting.

**Qualitative Evaluation.** In Fig. 2, we provide several example qualitative results, in which we can observe better results produced by our proposed method in comparison to VCNet [12]. For instance in the first row, though the eyes inpainted by VCNet look realistic, their appearance seems to be inconsistent with the whole face, while our model inpaints the eyes that are more visually coherent with the girl's face. Similar observation can be found in the fourth row (e.g. the eyes

**Table 2**. Ablation study for our model designs based on Places2 dataset (in which ImageNet dataset is used as the source of contaminant). "Trans. blocks" denotes the usage of transformer blocks in IIN.

| MPN | CPN | Trans. blocks | binary mask for $\mathbf{M}_{\mathrm{g}}$ | | | real-valued mask for $\mathbf{M}_{\mathrm{g}}$ | | |
|---|---|---|---|---|---|---|---|---|
| | | | PSNR↑ | SSIM↑ | LPIPS↓ | PSNR↑ | SSIM↑ | LPIPS↓ |
| | | | 23.08 | 0.8009 | 0.1574 | 21.08 | 0.7487 | 0.2275 |
| ✓ | | | 24.45 | 0.8436 | 0.1220 | 21.90 | 0.7798 | 0.2119 |
| ✓ | ✓ | | 24.65 | 0.8483 | 0.1146 | 22.20 | 0.7877 | 0.1966 |
| ✓ | | ✓ | 25.12 | 0.8531 | 0.1081 | 22.69 | 0.7952 | 0.1816 |
| ✓ | ✓ | ✓ | **25.22** | **0.8572** | **0.0998** | **22.74** | **0.7999** | **0.1742** |

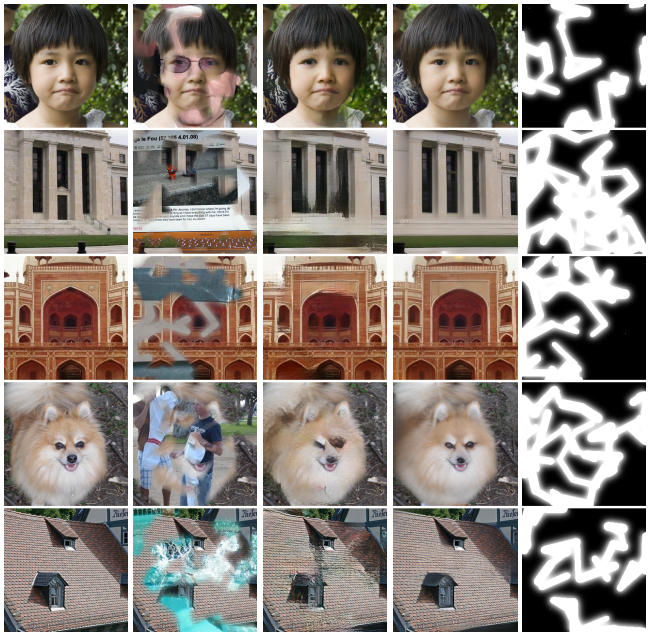


**Fig. 2**. Examples of qualitative results.

and the skin of the dog). Moreover, in the second row, we can see that our proposed model provides more structurally-reasonable inpainting result (e.g. straight pillars of the building) than VCNet. Overall, our model shows superior performance in recovering the corrupted regions while maintaining the holistic consistency with the uncorrupted regions.

**Test of Generalizability.** We further experiment to verify the generalizability of our proposed model: Inferring on the test samples generated by using different sources of contaminant (i.e. different from the one used to generate the training set for model learning). For instance, $\mathbf{N}$ is sampled from CelebA during training but turns to be sampled from Places2 during testing. Example results are shown in Fig. 3 where we can observe that our proposed model is still able to provide plausible estimates for $\mathbf{N}_{\mathrm{pred}}$ and reasonable inpainting result $\mathbf{I}_{\mathrm{pred}}$, thus validating the generalizability of our method.

**Ablation Study.** We use the Places2 dataset (in which the source of contaminant is the ImageNet dataset) to perform ablation studies on several model designs, including mask prediction network (MPN), contaminant prediction network (CPN), and adopting transformer blocks in the bottleneck of image inpainting network (IIN). Note that the model variant

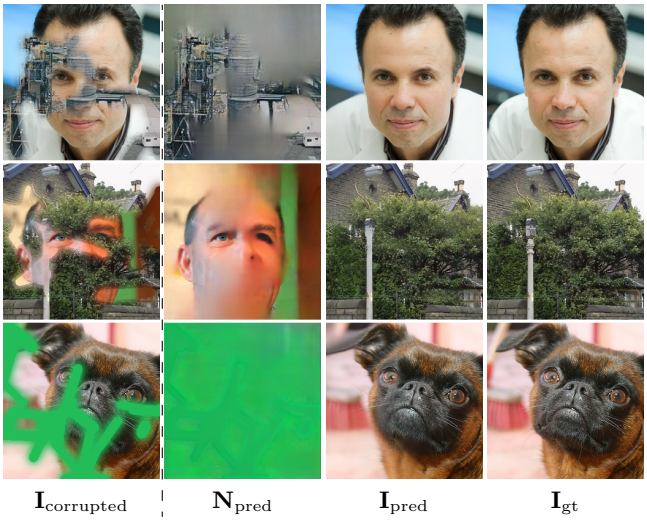


**Fig. 3**. Example results of the generalizability test.

with removing all these designs from our proposed method coincides with the coarse network of [16]. The results summarized in Table 2 reveal several contributions of our proposed model: 1) The mask $\mathbf{M}_{\mathrm{pred}}$ predicted by MPN is effective to guide the inpainting; 2) Our explicit estimate of contaminant (i.e. $\mathbf{N}_{\mathrm{pred}}$) further benefits the performance by making IIN more concentrate on inpainting (no matter whether the transformer blocks are adopted for IIN); 3) Introducing transformer blocks into IIN maintains the long-range semantic consistency between uncorrupted and corrupted regions and contributes to achieve the significant improvement. Moreover, we also experiment our model on using binary or real-valued mask for $\mathbf{M}_{\mathrm{g}}$ during data generation, where the results demonstrate a trend consistent with the aforementioned observations.

## 4. CONCLUSION

We propose a carefully-designed de-contamination model for the task of blind image inpainting in this work. The explicit estimate of the contaminant and its blending mask with the original image help eliminating the interference to the inpainting procedure, while the introduction of the transformer blocks further enhances the overall semantic consistency of the inpainted output. The extensive experiments not only verify the superiority of our method against both blind and non-blind inpainting methods, but also demonstrate its generalizability with respect to unknown sources of contaminant.

## 5. REFERENCES


[1] Nian Cai, Zhenghang Su, Zhineng Lin, Han Wang, Zhijing Yang, and Bingo Wing-Kuen Ling. Blind inpainting using the fully convolutional neural network. In *The Visual Computer*, 2017. 1

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 3

[3] S. Iizuka, E. Simo-Serra, and H. Ishikawa. Globally and locally consistent image completion. In *ACM Transactions on Graphics (TOG)*, 2017. 1

[4] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations (ICLR)*, 2018. 3

[5] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[6] Jingyuan Li, Ning Wang, Lefei Zhang, Bo Du, and Dacheng Tao. Recurrent feature reasoning for image inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 3

[7] Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *European Conference on Computer Vision (ECCV)*, 2018. 1

[8] Y. Liu, J. Pan, and Z. Su. Deep blind image inpainting. In *arXiv:1712.09078*, 2017. 1

[9] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z. Qureshi, and Mehran Ebrahimi. Edgeconnect: Generative image inpainting with adversarial edge learning. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2018. 1

[10] Fu-Jen Tsai, Yan-Tsung Peng, Yen-Yu Lin, Chung-Chi Tsai, and Chia-Wen Lin. Stripformer: Strip transformer for fast image deblurring. In *European Conference on Computer Vision (ECCV)*, 2022. 3

[11] Junke Wang, Shaoxiang Chen, Zuxuan Wu, and Yu-Gang Jiang. Ft-tdr: Frequency-guided transformer and top-down refinement network for blind face inpainting. In *IEEE Transactions on Multimedia (TMM)*, 2022. 1

[12] Yi Wang, Ying-Cong Chen, Xin Tao, and Jiaya Jia. Vcnet: A robust approach to blind image inpainting. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 3, 4

[13] Yi Wang, Xin Tao, Xiaojuan Qi, Xiaoyong Shen, and Jiaya Jia. Image inpainting via generative multi-column convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 1

[14] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. In *IEEE Transactions on Image Processing (TIP)*, 2004. 3

[15] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas Huang. Generative image inpainting with contextual attention. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[16] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas Huang. Free-form image inpainting with gated convolution. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 1, 2, 4

[17] Yanhong Zeng, Jianlong Fu, Hongyang Chao, and Baining Guo. Learning pyramid-context encoder network for high-quality image inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 3

[18] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[19] Haoru Zhao, Zhaorui Gu, Bing Zheng, and Haiyong Zheng. Transcnn-hae: Transformer-cnn hybrid autoencoder for blind image inpainting. In *ACM Conference on Multimedia (MM)*, 2022. 1

[20] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. Pluralistic image completion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[21] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2017. 3