Structure Preserving Image Inpainting Using Edge Priors with Contextual Attention

Ashish Kumar Singh(✉), Praveen Agrawal, Ankit Dhiman, Rishav Raj, Pankaj Kumar Bajpai, and Yash Harbhajanka

Samsung R&D Institute Bangalore, Bangalore, India
ashish.23ks@samsung.com, ashish23ks@gmail.com

Abstract. Deep learning techniques have produced plausible results for both regular and irregular masks for challenging task of image inpainting. Few approaches make use of extra information like edge priors for generator network, which preserves the structure which is blurry and distorted. On the other hand, certain approaches use surrounding patches to flow information in the missing regions, which in some scenarios can lead to erroneous output. Motivated by these approaches, we propose a three-stage architecture, which consists of an Edge Generator, followed by a Multi-Branch Image Generator and a Contextual Attention layer to generate high quality plausible patches in the input hole image. We evaluate the proposed architecture on the ICME 2019 Image Inpainting challenge and places2 dataset. The proposed method out-performs state of the art both quantitatively and qualitatively. This model can process rectangular holes at arbitrary locations.

Keywords: Image inpainting · Deep learning

1 Introduction

Image Inpainting refers to the process of filling missing portions in images in a way that the filled portions are consistent, perceptually plausible and merge smoothly with the whole image. The filled portions also need to preserve the structures in the image and be semantically accurate. Image Inpainting has a wide range of applications like restoring damaged or deteriorated portions of images and video frames, removing unwanted objects and modifying undesired regions of images.

Traditional techniques, in the field of Image Inpainting, use image statistics and low level features from the remaining portion of the image to fill the missing regions [3–6]. A lot of diffusion based approaches like [4,6] propose propagating the information from neighbouring background regions to the missing regions while some patch-based methods like [3,5] use patches from the available region of image to fill the missing portions of the image. These techniques work well for repetitive structures but lack the ability to capture high-level semantics for

R. V. Babu et al. (Eds.): NCVPRIPG 2019, CCIS 1249, pp. 100–110, 2020.
https://doi.org/10.1007/978-981-15-8697-2_9

complex structures and non-repetitive regions. The major drawback of such traditional methods is the inherent assumption that the low level features or the patches in the missing regions are present in the available image regions, and hence, such methods lack the ability to imagine novel structures in the missing region.

Recent deep learning approaches [11,14,17–19] have shown the capability of Generative Adversarial Networks (GANs) to learn relevant semantics to generate coherent structures in missing regions. An illustrative work based on this model is by Pathak et.al. [14], which uses encoder-decoder network, trained with reconstruction and adversarial loss for imagining contents in missing region. Both Pathak et. al. [14] and Yang et. al. [17] assumes 64 × 64 missing region in the centre of 128 × 128 image. Iijuka et. al. [11] uses global and local discriminator to generate consistent and coherent patches in the missing regions. These methods often produce blurry boundaries and texture artifacts primarily, because of the lack of structural information.

Recently, two-stage methods [13,15,19] are proposed to overcome the issue of blurry boundaries and texture artifacts. These methods try to recover structural information in first stage and generate finer details in second stage. Song et. al. [15] predicts semantic segmentation label in missing regions and then recovers finer details. However, different structures can be present in same semantic label region. Nazeri et. al. [13] first completes edges in missing region and then use this information to inpaint the missing region. But, the edges can only provide structural guidance for the inpainting step. Yu et. al. [19] tries to refine the coarse output of first stage with an attention layer in second stage. However, there is no attempt for preserving the structure. This attention layer flows the information from surrounding regions based upon matching patches of coarsely completed image from the previous stage. But it doesn't know if any region of the neighborhood needs to be ignored while inpainting in case they do not belong to the same object that is being in-painted. This leads to erroneous filling in the missing region. To mitigate this issue, we provide an edge map to contextual attention layer, which serves as a segmentation guide. This forces this layer to give weightage to edge of the image, thus improving structure as well as color and texture in the missing regions.

Based on these insights, we propose a novel three-stage model for image inpainting. The model uses edge priors for preserving structure and guiding refinement. The proposed network consists of an edge generator, coarse image generator and a refinement network. The edge generator completes edge information in the missing regions to generate edge priors for next steps. It is based on GAN framework that contain a generator and a discriminator. The coarse image generator uses this edge prior to produce structure preserved coarse image with holes filled. It is based on multi-channel network which uses convolutional kernels with different kernel size to provide better receptive field for preserving structures. The refinement network then takes this coarse image to produce meaningful textures in missing regions. It is based on two branch attention network that uses attention layer to generate high level texture. We use edge prior as guidance in the refinement network. The proposed method achieves significantly high quality inpainting results on ICME [1] and places2 [20] dataset and out-performs previous state-of-art methods.

Contributions of the proposed method are as follows:

- Novel three-stage model for image inpainting which uses edge priors for preserving structure and guiding refinement.
- Edge generator which completes missing edges to generate edge prior.
- Coarse image generator which uses edge prior to produce structure preserved coarse image with holes filled.
- Refinement network which uses edge prior to produce meaningful textures.
- We conducted qualitative and quantitative comparison with several state-of-the-art methods to show that proposed method can achieve competitive results.

2 Method

The architecture of proposed inpainting network is shown in Fig. 1. The proposed network consists of three parts: edge generator G_e, coarse image generator G_i, and refinement network G_r. The edge generator G_e generates edge map $\hat{E}$ by predicting the edges in missing regions. The coarse image generator G_i uses the information from the predicted edge map $\hat{E}$ to output coarse inpainted image $\hat{I}_c$. The refinement network G_r, refines the coarse image $\hat{I}_c$ using guidance from edge map $\hat{E}$ to output the final inpainted image $\hat{I}$.

2.1 Edge Generator

Similar to recent methods [13,15,19], we try to recover structural information in first stage before filling the missing regions. The edge generator G_e is used to fill edges in missing regions which preserve structures in the image. Let I_{gt} be the ground truth image and E_{gt} be the ground truth edge map of I_{gt}. The working of edge generator can be expressed as,

$$\hat{E} = G_e(I_{gray}, E_{in}, M) \tag{1}$$

where M is a binary mask in which 1 represents hole region and 0 represent non-hole region, $E_{in} = E_{gt} \odot (1 - M)$ is the edge map of the input image $I_{in} = I_{gt} \odot (1 - M)$ and I_{gray} is the grayscale image of I_{in}. Here, $\odot$ represents element-wise product.

Furthermore, we apply generative adversarial framework [8] to train the edge generator with the help of discriminator network D_e. The adversarial loss of the network can be written as,

$$\mathcal{L}^e_{adv} = \mathbb{E}[logD_e(E_{gt}, I_{gray})] + \mathbb{E}[log(1 - D_e(\hat{E}, I_{gray}))] \tag{2}$$

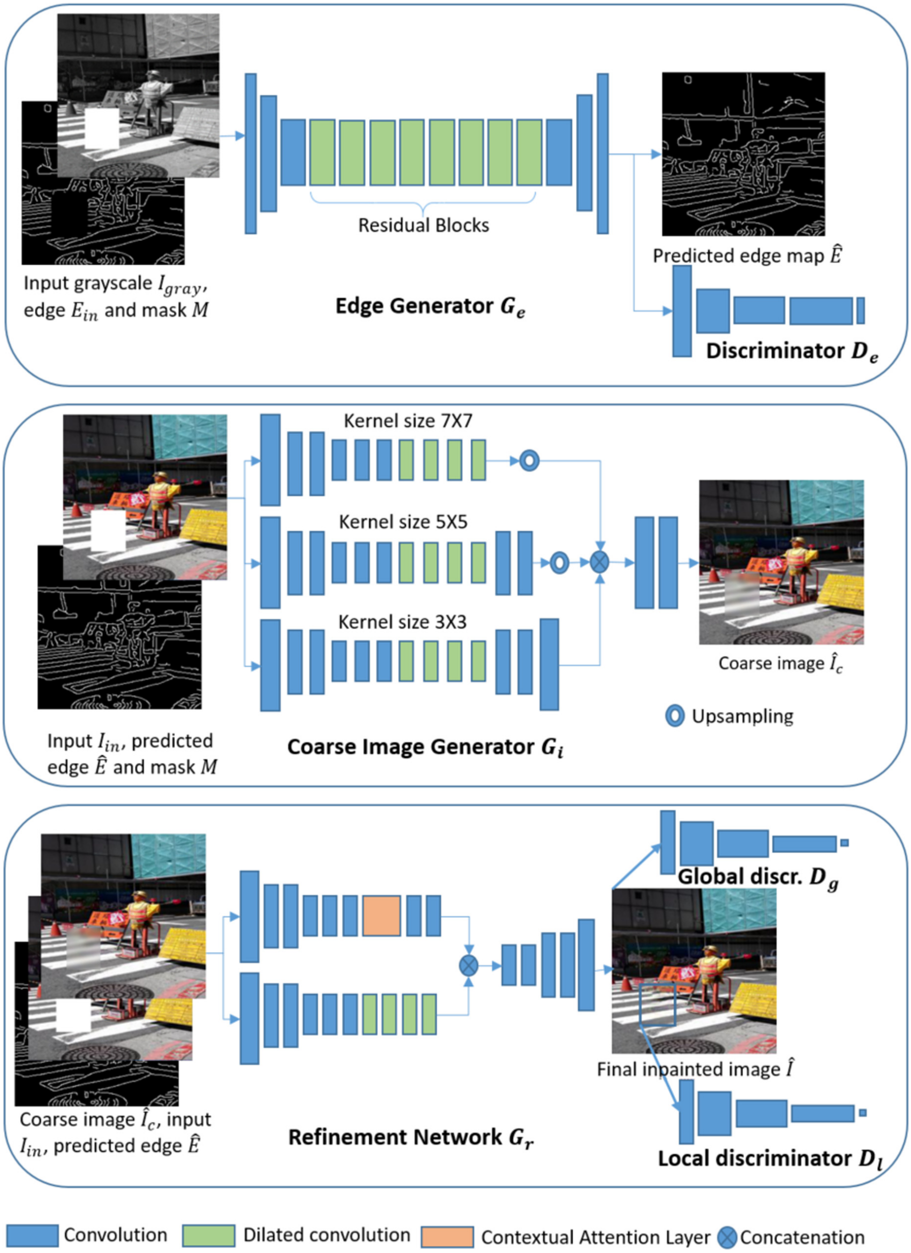
Residual Blocks
Predicted edge map $\hat{E}$
Input grayscale I_{gray}, edge E_{in} and mask M
Edge Generator G_e
Discriminator D_e
Kernel size 7X7
Kernel size 5X5
Kernel size 3X3
Coarse image $\hat{I}_c$
Upsampling
Input I_{in}, predicted edge $\hat{E}$ and mask M
Coarse Image Generator G_i
Global discr. D_g
Final inpainted image $\hat{I}$
Coarse image $\hat{I}_c$, input I_{in}, predicted edge $\hat{E}$
Refinement Network G_r
Local discriminator D_l
Convolution
Dilated convolution
Contextual Attention Layer
Concatenation

Fig. 1. Architecture of the proposed method

The generator G_e and discriminator D_e are trained jointly with the following optimization,

$$\underset{G_e}{min}\underset{D_e}{max} = \underset{G_e}{min}(\lambda^e_{adv}\underset{D_e}{max}(\mathcal{L}^e_{adv}) + \lambda_{fm}\mathcal{L}_{fm}) \tag{3}$$

where λ^e_{adv} and λ_{fm} are regularization parameter which are set to 1 and 10 respectively. We use feature matching loss $\mathcal{L}_{fm}$ which is computed on the activations of layers in discriminator D_e as proposed in [13].

2.2 Coarse Image Generator

After getting the edge map $\hat{E}$, the coarse image generator G_i is used to generate coarse image $\hat{I}_c$. The processing of coarse image generator can be expressed as

$$\hat{I}_c = G_i(I_{in}, \hat{E}, M) \tag{4}$$

Pixels near hole boundaries have less ambiguity than pixels that are far from hole boundaries. So it is sensible to use different weights for these pixels when calculating loss. Similar weight ideas are explored in [14,19]. Inspired by [19], we use spatially discounted reconstruction loss using a weight mask M_w. The weight of each pixel in mask is computed as γ^l, where l is the distance of pixel from the nearest non-hole pixel. γ is set to 0.99. We calculate spatial discounted $L1$ loss on the output coarse image $\hat{I}_c$ as following

$$\mathcal{L}^{coarse}_{l_1,hole} = \left\|\hat{I}_c \odot M_w - I_{gt} \odot M_w\right\|_1 \tag{5}$$

$$\mathcal{L}^{coarse}_{l_1,non-hole} = \left\|\hat{I}_c \odot (1-M) - I_{gt} \odot (1-M)\right\|_1 \tag{6}$$

where $\mathcal{L}^{coarse}_{l_1,hole}$ and $\mathcal{L}^{coarse}_{l_1,non-hole}$ are the L_1 loss in hole and non-hole regions respectively. Spatial discounting loss is used in hole region L_1 loss calculation using mask M_w. Coarse image generator G_i is trained in conjunction with refinement network G_r.

2.3 Refinement Network

The refinement network G_r takes the coarse image $\hat{I}_c$, edge map $\hat{E}$ and outputs the final image $\hat{I}$. The working of refinement network can be written as

$$\hat{I} = G_r(\hat{I}_c, \hat{E}, M) \tag{7}$$

We train the refinement network G_r along with coarse image generator G_i following global and local Wasserstein GANs framework [2,9] using local and global discriminator D_l and D_g respectively. Inspired by [9,19] we use gradient penalty loss to both global and local outputs to enforce structural consistency.

Similar to Eqs. 5 and 6 we calculate $\mathcal{L}^{refine}_{l_1,hole}$ and $\mathcal{L}^{refine}_{l_1,non-hole}$ for final image $\hat{I}$. The full L_1 losses is computed as

$$\mathcal{L}_{l_1,hole} = \mathcal{L}^{refine}_{l_1,hole} + \lambda_{coarse}\mathcal{L}^{coarse}_{l_1,hole} \tag{8}$$

$$\mathcal{L}_{l_1,non-hole} = \mathcal{L}^{refine}_{l_1,non-hole} + \lambda_{coarse}\mathcal{L}^{coarse}_{l_1,non-hole} \tag{9}$$

Here, $\lambda_{coarse} = 1.2$ is a regularization parameter, $\mathcal{L}_{l_1,hole}$ and $\mathcal{L}_{l_1,non-hole}$ corresponds to the reconstruction loss in hole and non-hole region respectively for training G_i and G_r together. The adversarial loss including the gradient penalty [9] terms can be written as

$$\mathcal{L}^{local}_{adv} = \mathbb{E}[D_l(\hat{I}_{c,local})] - \mathbb{E}[D_l(I_{gt,local})] + \lambda_{gp}\mathbb{E}[(\left\|\nabla D_l(\hat{I}_{c,local})\right\|_2 - 1)^2] \tag{10}$$

$$\mathcal{L}^{global}_{adv} = \mathbb{E}[\lambda_{global}D_g(\hat{I}_c)] - \mathbb{E}[\lambda_{global}[D_g(I_{gt})] + \lambda_{gp}\mathbb{E}[(\left\|\nabla D_g(\hat{I}_c)\right\|_2 - 1)^2] \tag{11}$$

Here, $\lambda_{global} = 1$ and $\lambda_{gp} = 10$ are regularization parameters. $\hat{I}_{c,local}$ and $I_{gt,local}$ are the cropped images corresponding to the hole regions in mask M of $\hat{I}_c$ and I_{gt} respectively. We also compute well known Perceptual loss L_{perc} and style loss L_{sty} [7,12] using pretrained VGG weights. The full objective function of the W-GAN framework can be expressed as

$$\mathcal{L} = \mathcal{L}^{local}_{adv} + \mathcal{L}^{global}_{adv} + \lambda_h\mathcal{L}_{l_1,hole} + \lambda_{nh}\mathcal{L}_{l_1,non-hole} + \lambda_{perc}L_{perc} + \lambda_{sty}L_{sty} \tag{12}$$

For conducting experiments, we choose $\lambda_h = 6$, $\lambda_{nh} = 1$, $\lambda_{perc} = 0.5$ and $\lambda_{sty} = 250$.

3 Architecture and Training

The edge generator G_e follows encoder decoder architecture with residual blocks [10] in middle to process features. G_e contains an encoder that down-samples twice, followed by eight residual blocks and decoder that up-samples to the original size. The edge discriminator D_e is a simple five layer convolutional network that outputs whether the input edge image is real or fake. This architecture is similar to that proposed by Nazeri et al. [13]. Canny edges are used to generate incomplete edge input E_{in} which are fed to generator to hallucinate edges in missing areas. We used pre-trained weights provided by [13] for edge generator G_e to generate structural information in the missing regions.

The coarse image generator G_i contains three parallel branch with different size convolutional kernels to get information from different receptive fields. It also contains a merge layer to merge outputs of three branches together. This architecture is inspired by [16]. A five layer convolutional network based local discriminator D_l is used which takes only the mask regions input and outputs whether an image is real or not. The refinement network G_r contains two parallel branch and a merge layer to merge the outputs together. One of the branch uses Contextual Attention layer that uses patches from non-hole regions in image as kernels. The use of Contextual Attention layer is inspired by [19]. A five layer CNN architecture based global discriminator D_g is used which takes full image input and outputs whether an image is real or not. Figure 1 shows the proposed architecture, their sample inputs and outputs.

The proposed model is trained on places2 [20] and ICME [1] dataset. The network is trained using 256 × 256 images with batch size 16. The hole generated in mask M is rectangular in shape. The height and width of the hole is randomly selected from the range (32, 64) pixels. The generated hole is placed randomly to form the mask M with hole. Adam Optimizer is used with learning rate of 10^{-5}. The proposed edge generator (G_e) has a total of 10.8M parameters and takes 41.5 MB space. The proposed coarse and refinemnet generator ($G_i + G_r$) has 14.7M parameters and takes 56.3MB space. Pytorch framework is used for coding and training models. The models are trained on hardware with CPU Intel(R) Xeon(R) CPU E5- 2697 v3 (2.60 GHz) and 4 GPUs GTX 1080 Ti.

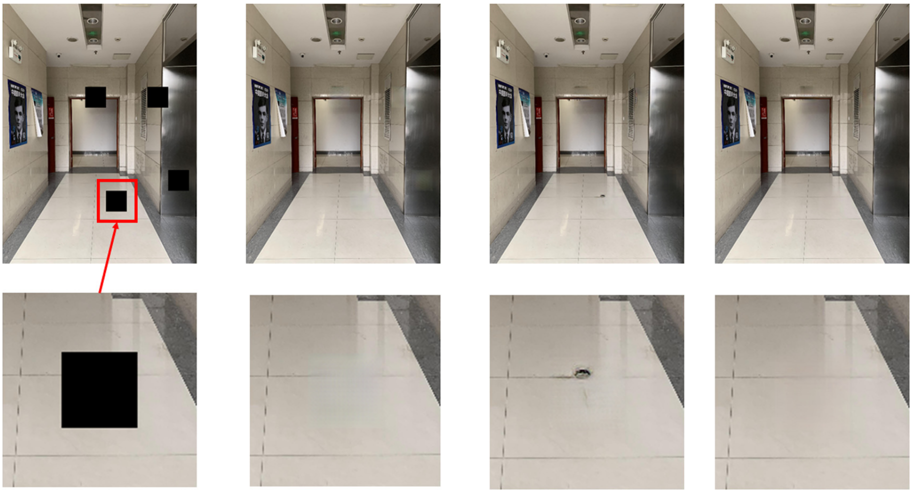

Fig. 2. Sample from ICME test dataset [1]: (left to right) damaged image, EdgeConnect [13], Contextual Attention [19], proposed approach. Bottom row: zoomed in version of a patch. In zoomed patch, the proposed method completes the line partitioning the tiles while keeping the two well segmented regions to be inpainted without any overflow from one region to the other. Whereas, Contextual attention method [19], fails to correctly complete the line as it has no information of the underlying structure.

4 Experiments and Results

We evaluated the proposed inpainting method on two datasets: Places2 [20] test dataset and ICME 2019 Inpainting challenge's test dataset [1]. We compare the output of proposed network with state-of-the-art methods [13,19].

For comparison on Places2 [20] test dataset, we used the pre-trained weights provided by authors of [13,19], trained on Places2 dataset. For comparing the results, we used images of resolution 256 × 256 with a hole at the image center of resolution 64 × 64. For comparison on ICME dataset, we fine-tune the pre-trained weights (trained on Places2 [20]) provided by both networks on ICME data [1]. We used regular masks with 4 rectangular holes provided in ICME [1] dataset. All images were resized to 256 × 256 for training and testing.

All the results reported are direct outputs from the trained models. No post-processing step is involved while reporting the results.

Qualitative Comparison. In Fig. 2 and 3, we show that the proposed three-stage model generates superior results than EdgeConnect [13] and Contextual Attention [19]. Proposed method uses edge as a prior information for contextual attention layer, this aids the contextual attention layer to fill right texture in the missing region.

Table 1. Results of PSNR and SSIM on ICME test dataset

Method	PSNR	SSIM
EdgeConnect [13]	31.3076	0.9781
Contextual attention [19]	30.7931	0.9784
Proposed method	**31.9059**	**0.9791**

Quantitative Comparison. We report peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [21] on ICME test data [1]. As shown in Table 1, proposed method outperforms the other two methods in the reported metrics. For comparison on Places2 [20] dataset, we report PSNR and SSIM metric on both the patch as well as the full image. As shown in Table 2, proposed method outperforms the other state-of-the-art methods.

Table 2. Results of PSNR and SSIM on Places2 test dataset

Method	Inpainted patch		Full image	
	PSNR	SSIM	PSNR	SSIM
EdgeConnect [13]	16.6757	0.3099	28.7107	0.9563
Contextual attention [19]	15.8059	0.2922	27.7868	0.9530
Proposed method	**16.7752**	**0.3181**	**28.8067**	**0.9569**

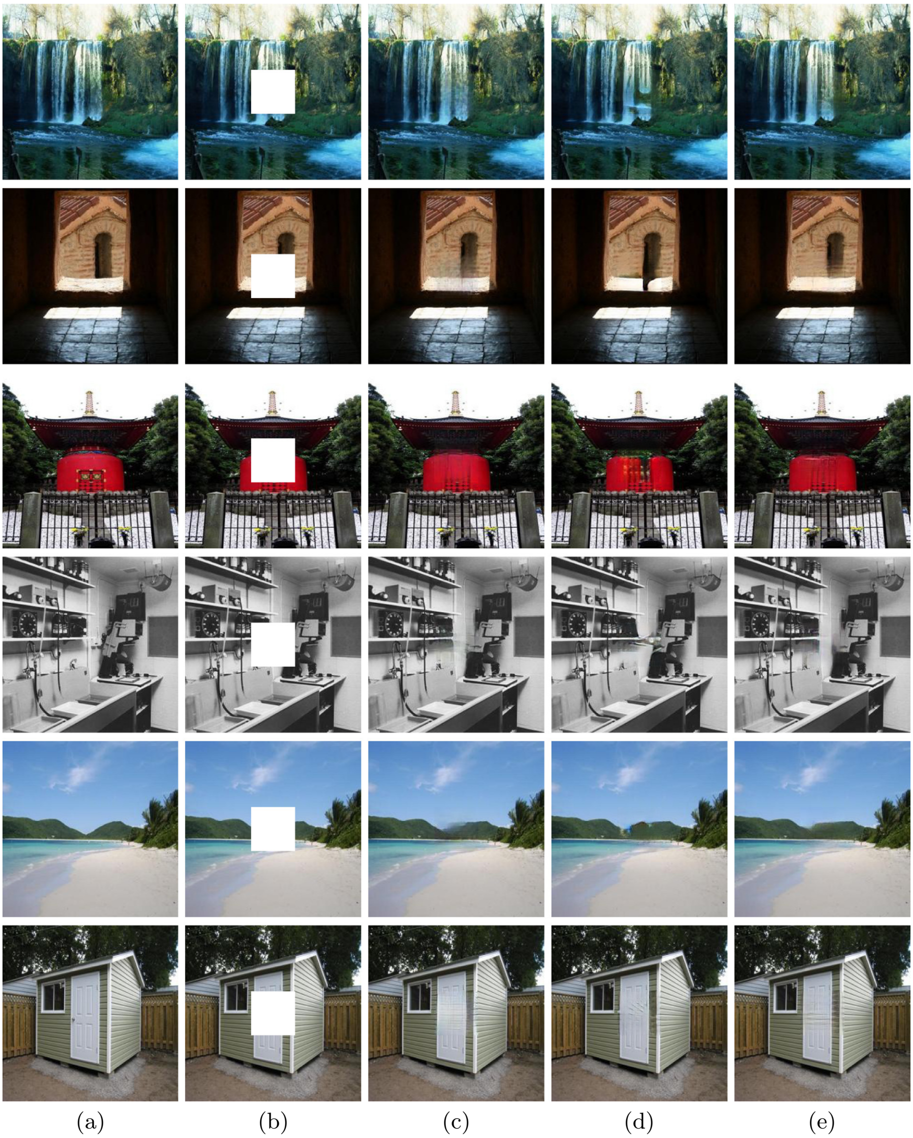

Fig. 3. Results on Places2 test dataset (a) ground truth image, (b) damaged image (c, d, e) Results (c) EdgeConnect [13], (d) Contextual Attention [19], and (e) Proposed method

5 Conclusion

We proposed a three-stage image completion network, which comprises of an edge generator, a multi-branch coarse image generator and refinement network with contextual attention layer. Also, we demonstrated how the edge information can be used to improve results of Contextual Attention Network. The proposed method outperforms state of the art methods on both qualtative and quantitative evaluation. The experimental results obtained, shows the feasibilty of the proposed method. For future work, we plan to experiment on better prior than Edges. We have observed that in textured regions, because of the spurious nature of output from edge completion network, output from the proposed method is not as expected. Furthermore, we would also like to extend this method for high-resolution images.

References

1. ICME 2019: Learning-Based Image Inpainting Challenge (2019). https://icme19inpainting.github.io/
2. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein GAN. arXiv e-prints arXiv:1701.07875, January 2017
3. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: PatchMatch: a randomized correspondence algorithm for structural image editing. ACM Trans. Graph. (ToG) **28**, 24 (2009)
4. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2000, pp. 417–424. ACM Press/Addison-Wesley Publishing Co. (2000)
5. Darabi, S., Shechtman, E., Barnes, C., Goldman, D.B., Sen, P.: Image melding: combining inconsistent images using patch-based synthesis. ACM Trans. Graph. **31**(4), 82:1–82:10 (2012)
6. Efros, A.A., Freeman, W.T.: Image quilting for texture synthesis and transfer. In: Proceedings of SIGGRAPH 2001, pp. 341–346 (2001)
7. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2414–2423. IEEE Computer Society, June 2016
8. Goodfellow, I.J., et al.: Generative adversarial networks. arXiv e-prints arXiv:1406.2661, June 2014
9. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.: Improved training of Wasserstein GANs. arXiv e-prints arXiv:1704.00028, March 2017
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arXiv e-prints arXiv:1512.03385, December 2015
11. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Globally and locally consistent image completion. ACM Trans. Graph. (Proc. SIGGRAPH 2017) **36**(4), 107:1–107:14 (2017)
12. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. arXiv e-prints arXiv:1603.08155, March 2016
13. Nazeri, K., Ng, E., Joseph, T., Qureshi, F., Ebrahimi, M.: EdgeConnect: generative image inpainting with adversarial edge learning. arXiv preprint arXiv:1901.00212 (2019)

14. Pathak, D., Krähenbühl, P., Donahue, J., Darrell, T., Efros, A.: Context encoders: feature learning by inpainting (2016)
15. Song, Y., Yang, C., Shen, Y., Wang, P., Huang, Q., Kuo, C.J.: SPG-net: segmentation prediction and guidance network for image inpainting. CoRR abs/1805.03356 (2018)
16. Wang, Y., Tao, X., Qi, X., Shen, X., Jia, J.: Image inpainting via generative multi-column convolutional neural networks. arXiv e-prints arXiv:1810.08771, October 2018
17. Yang, C., Lu, X., Lin, Z., Shechtman, E., Wang, O., Li, H.: High-resolution image inpainting using multi-scale neural patch synthesis. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4076–4084, July 2017
18. Yeh, R.A., Chen, C., Yian Lim, T., Schwing, A.G., Hasegawa-Johnson, M., Do, M.N.: Semantic image inpainting with deep generative models. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017
19. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Generative image inpainting with contextual attention. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5505–5514 (2018)
20. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: a 10 million image database for scene recognition. IEEE Trans. Pattern Anal. Mach. Intell. **40**(6), 1452–1464 (2017)
21. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)