Performance Evaluation of Image Inpainting Algorithms Proposed in Pluralistic Image Completion Paper

Team members: Huong Doan and Yunting Chiu

Basically, image inpainting is to recover the damaged image or fill out a missing part of an image. There are a lot of existing image inpainting methods which are published with or without codes. Those methods will have their own advantages and disadvantages. In addition, those methods usually work well on the datasets which were used for the experiments on the paper but it does not guarantee that those methods can work well for other datasets as well. Several related readings are Texture Memory-Augmented Deep Patch-Based Image Inpainting[5], Image Fine-grained Inpainting[4] and Generative Adversarial Networks[3].

In our study, we choose an existing method which we are interested in and apply it to the interesting datasets we choose to observe its performance as well as its limitation if there is any. We choose the method proposed in the **Pluralistic Image Completion**[6] paper to investigate the two datasets taken from Flickr-Faces-HQ Dataset (FFHQ)[2] and Cifar-10[1]. The main challenging parts are to figure out how to run the github code using our datasets and how to evaluate the performance of our experiments. For the experiments, we are planning to change the number of images in the train file to observe whether the results are different. However, our training images (65536) are greater than the one the authors used for their experiments (24183 for training and 2824 for testing). With the limited computer environment and resources, we cannot comprehensively finish the training process. Despite the fact that the training process is only at epoch 41(epoch is the number of passes of the entire training dataset), the output is satisfactory so far. As a result, one image will generate 49 different types of faces so we believe that the result can fool people. The limitation, we think it could be, is the running time and that is the more images in the training or testing set, the longer the running time of the code will be. The solutions should either decrease the number of input images or increase the training time. We will visualize the recovered images and the original images without missing parts. The expected results are recovered images but the methodology of the paper does not guarantee that those images exactly look like the original images without missing parts. In order to evaluate the performance of the model proposed in the paper, we will use total variance loss (TV loss - the lower the better), Structural Similarity Index (SSIM - the higher the better) and Peak Signal-to-Noise Ratio (PSNR - the higher the better). Total variation loss measures how much noise is in the images, while SSIM measures the similarity between two given images. Besides, PSNR is to compare the output images (the inpainting images) to the input images (the images with holes or masks) with the maximum possible power.

In the Pluralistic Image Completion paper, the author proposed the methodology of generating multiple and diverse plausible solutions for image completion. The architecture of the method is shown below.

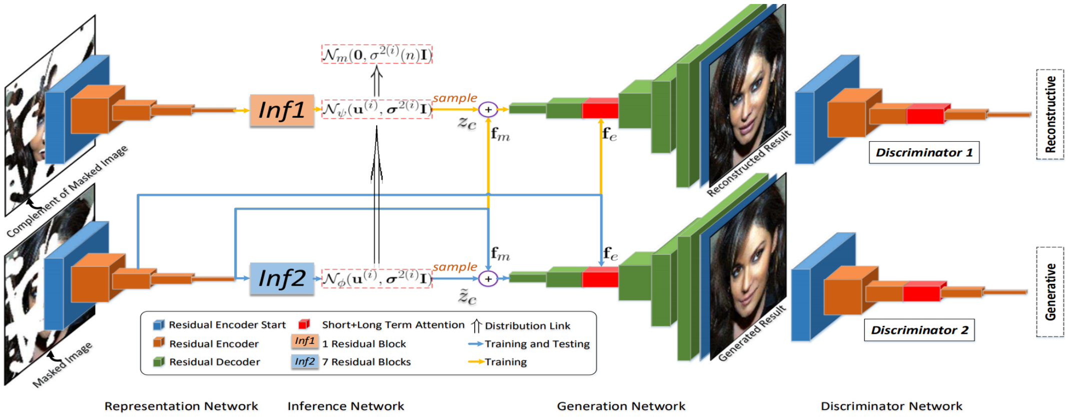

Figure 3. Overview of our architecture with two parallel pipelines. The **reconstructive** pipeline (yellow line) combines information from $\mathbf{I}_m$ and $\mathbf{I}_c$, which is used only for training. The **generative** pipeline (blue line) infers the conditional distribution of hidden regions, that can be sampled during testing. Both representation and generation networks share identical weights.

This neural network consists of two parallel pipelines. The yellow line (reconstructive) merges data from Im and Ic, which are only used for training purposes. The blue (generative) pipeline estimates the conditional distribution of hidden regions, which can then be sampled during testing, that is, Ig = {Ic, Im}.

- Ig is the original image, and Im is the masked image. The method is mapping Ig to Im.
- Define Ic as the converse of Im, which is constructed from the masked image.
- This paper final goal is to take sample from p(Ic|Im) to recover images.

References

1. Cifar-10 dataset (https://www.cs.toronto.edu/~kriz/cifar.html)
2. Flickr-Faces-HQ Dataset (FFHQ) dataset (https://github.com/NVlabs/ffhq-dataset)
3. Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial networks. *arXiv preprint arXiv:1406.2661.*
4. Hui, Z., Li, J., Wang, X., & Gao, X. (2020). Image fine-grained inpainting. *arXiv preprint arXiv:2002.02609.*
5. Xu, R., Guo, M., Wang, J., Li, X., Zhou, B., & Loy, C. C. (2020). Texture Memory-Augmented Deep Patch-Based Image Inpainting. *arXiv preprint arXiv:2009.13240.*
6. Zheng, C., Cham, T. J., & Cai, J. (2019). Pluralistic image completion. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 1438-1447).*