Team Members
Grigor Nalbandyan
Arpen Matinyan
Levon Avetisyan
Inst. Aram Keryan

Deep Learning Final Project
Image Inpainting

Abstract

This paper aims to reproduce the results of the PEPSI paper. Pepsi incorporates the encoder-decoder approach. It has two generator decoder paths - coarse path and refinement path. Compared to other methods, both paths share the same decoder instead of different decoders for each path. This way PEPSI decreases the number of model's parameters. It also has the CAM - contextual attention module which helps the model to understand how to reconstruct the foreground - the masked part, from background - the non masked part. Conventional methods of image inpainting have two discriminators: local and global, to assess the quality of generated images, but PEPSI has one discriminator - region ensemble discriminator (RED) which unifies those two discriminators into one. Main advantage of RED is that it can handle not only rectangular masks but any type of masks. In the scopes of this project we have replicated the PEPSI architecture, changed it in several ways and improved it by reducing the number of parameters, whilst having similar results.

1. Introduction

Assume you have an old image with some deteriorated or missing parts, that you want to reconstruct, or maybe you have an unwanted object in the image that you want to remove. Today a technique called image inpainting is used for solving such kinds of problems. Image inpainting is the process of reconstructing missing parts of an image, so that the observers are unable to tell that these regions have undergone restoration. In this project we will try to implement image inpainting using a CNN and GAN-based architecture.

2. Related Work

Image inpainting techniques have two main approaches: *image processing* based and *learning* based. Image processing based approaches were used in early stages of image inpainting solutions. The first attempts were mainly based on copying background patches from

the image and reconstructing the damaged parts. Barnes et al. (2009) [2] suggest an algorithm of quickly finding approximate nearest-neighbor matches between image patches. The results were promising, however very limited, because this algorithm assumes that all the necessary parts are available in the background, which is not true in most cases.

Further advances of CNNs led to simple encoder-decoder techniques which enabled to have significantly better results than pure image processing. Later progress in GANs led to even more natural inpainting results as GANs could assess both the quality of generated part and overall image coherency during training. In Yu et al. [4] CAM and coarse network were used for image inpainting, with local and global discriminators, but this method worked only on rectangular masked images. Liu et al in [6] make use of partial convolutions to solve image inpainting and were the first to handle irregular shape masks. Later, Yu et al. [5] used gated convolutions and more complex network architecture similar to PEPSI, with coarse path, two refinement networks one of which had CAM. Yu et al. [5] method had Spectral-Normalized Markovian Discriminator similar to RED in the sense of unifying the local and global discriminators, but it classified each neuron of final feature map independently. This method proved to be especially helpful for free form masks and had comparable results with [6]. PEPSI++ or Diet-PEPSI is another network from the same authors of PEPSI (Shin et al [7]) where they propose decrease of number of parameters of inpainter. They observed that nearly 67% of all weights in inpainter are in dilated convolutions of encoder and proposed to have one tensor W and shift and scale to get weights for each dilated convolutional layer. This let them to decrease the inpainter's parameters by 30% with insignificant loss in performance.

3. Dataset and Features

In the scopes of this project we are going to use a dataset of face images - CelebA-HQ dataset, which is a dataset of 256x256 high resolution celebrity faces. Although we choose this particular dataset, the overall approach is applicable for any type of data. In the official paper, 30000 data samples were used for training the model. We chose this dataset to compare our results with the results in the official paper. Below you can find some examples from the dataset.

4.Methods

As shown in Figure 1, the overall architecture of Pepsi is conventional encoder-decoder. First, the masked image is downsampled-encoded by an CNN based encoding network and

then upsampled-decoded by the CNN based decoder network and the output is the inpainted result.

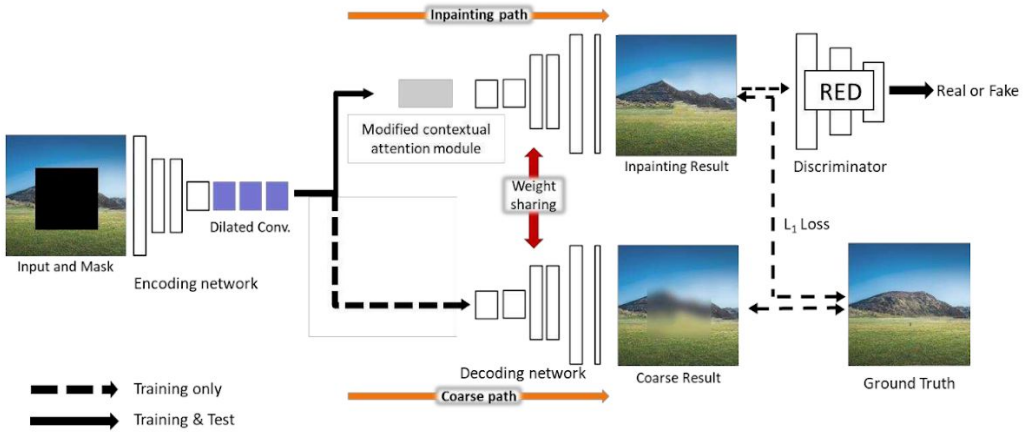

Figure 1. An architecture of PEPSI. The coarse path and inpainting path share their weights to improve each other. The coarse path is trained only with the ℓ_1 reconstruction loss while the inpaiting path is trained with both of ℓ_1 and adversarial loss

Type	Kernel	Dilation	Stride	Outputs
Convolution	5×5	1	1×1	32
Convolution	3×3	1	2×2	64
Convolution	3×3	1	1×1	64
Convolution	3×3	1	2×2	128
Convolution	3×3	1	1×1	128
Convolution	3×3	1	2×2	256
Dialated convolution	3×3	2	1×1	256
Dialated convolution	3×3	4	1×1	256
Dialated convolution	3×3	8	1×1	256
Dialated convolution	3×3	16	1×1	256

a) Encoder

Type	Kernel	Dilation	Stride	Outputs
Convolution $\times 2$	3×3	1	1×1	128
Nearest Neighbor ($\times 2 \uparrow$)	-	-	-	-
Convolution $\times 2$	3×3	1	1×1	64
Nearest Neighbor ($\times 2 \uparrow$)	-	-	-	-
Convolution $\times 2$	3×3	1	1×1	32
Nearest Neighbor ($\times 2 \uparrow$)	-	-	-	-
Convolution $\times 2$	3×3	1	1×1	16
Convolution (Output)	3×3	1	1×1	3

b) Decoder

The result of the encoder network is passed through the decoder twice, by coarse path and by inpainting part. During the coarse path encoded image is passed through the decoder to produce a coarse result. During the inpainting part the encoded image is passed through the contextual attention module (CAM) and then decoder to produce the inpainted result. During both passes the decoder network is the same, which allows to reduce memory and computational usage compared to two distinct decoders.

If we omit the CAM and second decoding part, the network may not be able to generate high quality inpainted results, as it won't understand the relation between background and foreground (masked) regions. CAM divides background and foreground into 3x3 patches and calculates similarity score between them. Similarity score which is euclidean distance, is a scoring mechanism to understand which background patches are important for masked part reconstruction. CAM uses these similarity scores as weights and reconstructs foreground as a weighted sum of background patches.

In general image inpainting models have two discriminators - local and global. Local discriminator assesses the quality of the generated masked part, the global discriminator assesses overall coherency of image and the generated masked part. PEPSI unifies those two discriminators into one - region ensemble discriminator (RED). RED is a sequence of convolutional filters, and in the end fully connected layers are applied to each pixel of the last feature map depthwise. Each fully connected network will classify whether each receptive field

is real or not. In this way RED can be a universal discriminator for any type of mask, compared to local and global discriminators which are used with square masks. Spectral normalization is applied to RED, which is known to stabilize the training of GANs.

Type	Kernel	Stride	Outputs
Convolution	5×5	2×2	64
Convolution	5×5	2×2	128
Convolution	5×5	2×2	256
Convolution	5×5	2×2	256
Convolution	5×5	2×2	256
Convolution	5×5	2×2	512
FC	1×1	1×1	1

c) RED

Training of Pepsi is done by jointly optimizing coarse path, inpainting paths and RED. For coarse path, L1 loss is calculated between constructed image and the ground truth. Loss of inpainting path is sum of L1 loss between output and ground truth and discriminator loss. The weight of coarse path loss is decreased during training and coarse path wont be used during testing.

$$L_G = \frac{\lambda_{\text{i}}}{N} \sum_{n=1}^{N} \|X_{\text{i}}^{(n)} - Y^{(n)}\|_1 - \lambda_{\text{adv}} E_{x \sim P_{X_{\text{i}}}}[D(x)], \quad L_{\text{total}} = L_G + \lambda_{\text{c}}\left(1 - \frac{k}{k_{\text{max}}}\right) L_C,$$

Loss of RED is hinge loss.

$$\begin{aligned} L_D = & -\mathbb{E}_{(x,y)\sim p_{data}}[\min(0, -1 + D(x,y))] \\ & -\mathbb{E}_{z\sim p_z, y\sim p_{data}}[\min(0, -1 - D(G(z),y))] \end{aligned}$$

Although PEPSI has a shared decoder and reduces the number of model's parameters, the number of parameters is still high. To cope with this problem, we replaced the standard convolutions in PEPSI by depth wise separable convolutions (Howard et al., [3]). The difference in number of parameters is presented below.

Parameters in each module with standard convolutions	Parameters in each module with depthwise separable convolutions
Encoder 2,937,184	Encoder 342,076
ContextualBlock 131,328	CAM 131,328
Decoder 588,435	Decoder 73,171
Discriminator Red 7,592,080	Discriminator Red 338,014
Total 11,249,027	Total 884,589

By changing the original convolutions to depthwise convolutions we decreased the number of parameters by about 12 times.

5.Results

We have contacted the authors of the PEPSI and it turned out that PEPSI was trained for 14 days. Despite the decrease of number of parameters, training time did decrease but not

proportionally. We were not able to train the model on full data due to limits of computational power. We used a batch of 16 images and tried to overfit the batch to see whether the model is able to overfit a single batch. The results of the model trained to predict the masked part with 128x128 center mask are presented below.

The whole process of the model's training can be seen with these pictures provided in the following link:https://drive.google.com/open?id=1bR0XV6SbpCz2SFQOr7ritQGOWwIudaop.

6.Conclusion

Image inpainting is an active area of research. The project taught us the whole process of machine learning project beginning from research of the area, data gathering, implementation of the paper, getting and interpreting the results. PEPSI is a mix of CNNs and GANs and incorporates main ideas of current image inpainting solutions. Hence, we are now fully informed on the current state of both image inpainting solutions and usage of GANs in general. Due to limited computational power, we couldn't fully reproduce the results in the paper. Application of depthwise separable convolutions helped to decrease the number of parameters and partially solve the problem of limited computational power. We would like to train the model fully in future and if reached higher results, our model will be applicable to any system with hardware constraints.

7.Contributions

Mostly the project work was done during group meetings. More specifically the work was divided as follows.

Grigor Nalbandyan: Has worked on the main part of implementation and introduced the notion of depth wise separable convolutions to the code.

Arpen Matinyan: Has worked on papers and theories comparison, finding out the best way to implement GANs in our model.

Levon Avetisyan: Has worked on testing and exploration of parameters, as well as finding techniques to use non-default mask construction functions.

References

1. M. Sagong, Y. Shin, S. Kim, S. Park, S. Ko. "PEPSI : Fast Image Inpainting with Parallel Decoding Network." http://openaccess.thecvf.com/content_CVPR_2019/papers/Sagong_PEPSI__Fast_Image_Inpainting_With_Parallel_Decoding_Network_CVPR_2019_paper.pdf
   PEPSI implementation. https://github.com/Forty-lock/PEPSI-Fast_image_inpainting_with_parallel_decoding_network/blob/master/main.py
2. C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," https://gfx.cs.princeton.edu/pubs/Barnes_2009_PAR/patchmatch.pdf
3. A. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam. "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications" https://arxiv.org/abs/1704.04861
4. J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention" https://arxiv.org/pdf/1801.07892.pdf
5. J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution" https://arxiv.org/pdf/1806.03589.pdf
6. G. Liu, F. Reda, K. Shih, T. Wang, A. Tao, B. Catanzaro. "Image Inpainting for Irregular Holes Using Partial Convolutions" https://arxiv.org/pdf/1804.07723.pdf
7. Y. Shin, M. Sagong, Y. Yeo, S. Kim, S. Park, S. Ko. "PEPSI++: Fast and Lightweight Network for Image Inpainting" https://arxiv.org/pdf/1905.09010.pdf