Inpainting with SinGAN: Single Generative Adversarial Nets
Project Proposal
Deep Learning course MVA

Megi Dervishi
École Normale Supérieure
45 Rue d'Ulm
megi.dervishi@ens.fr

Abstract

This project explores SinGAN (single generated adversarial network) which was first introduced by [1] and gives very impressive results in image manipulation. In this project I want to experiment how we can use SinGAN to perform inpainting on a masked image and what are its limitations.

1. Motivation

GANs (Generative Adversarial Networks) in the last few years have attracted a lot of attention in computer vision thanks to their ability to generate realistic images [5-10] and hence be applicable in many different image manipulation tasks such as paint-to-image, harmonization or image editing. Unfortunately, GANs are very hard to train and hence require a great amount of training images in order to learn a good representation of features, especially when there are complicated images. Therefore, the paper [1] introduced SinGAN, a new approach which trains on a single image, using a pyramid of fully convolutional lightweight GANs each responsible for capturing the distribution of patches at different scales (see Figure 1). Image inpainting is an essential functionality in many imaging and graphics applications such as: object removal, image restoration, image manipulation, re-targeting, compositing and image-based rendering [7, 8, 9]. Paper [5] presents a review of past and modern inpainting techniques.

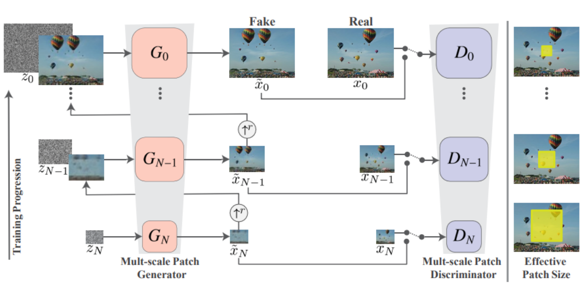

Figure 1: SinGANs architecture.

1.1. Problem Definition

This project will consist in understanding and reproducing some of the main results of the SinGAN initial paper [1]. After which I would like to use SinGAN in order to perform inpainting techniques. Hence study different ways of performing the inpainting, such as regular vs. irregular holes, 1 vs. 2 vs. n-holes, one-step or progressive inpainting and comparing it to the current state of the art [5].

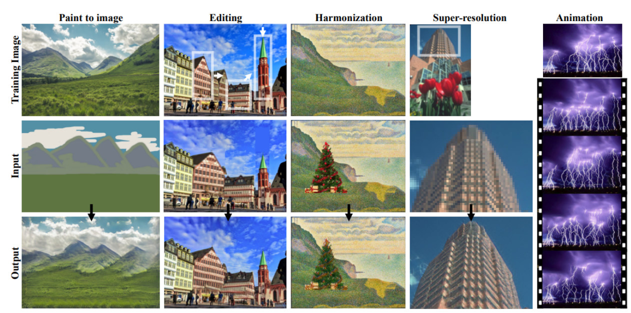

Figure 2: Image manipulation using SinGAN.

2. Methodology

The applications of the SinGAN in the paper consist of Paint-to-Image, Harmonization, Editing, Super-resolution and Animation from a single image which are displayed in Figure 2. Note that all tasks are achieved with the same generative network, the only difference comes from the training image and the input image. I will start by reproducing the results obtained in the original paper [1] and then proceed to implement in-painting.

2.1. In-Painting

First step before doing in-painting is preprocessing the data (i.e., masking them), as I would like to study the impact of holes on the performance of SinGAN, in terms of:

- Shape (bounding box, fitted object mask)
- Size
- Number
- Irregularity of holes
- Colors

In order to mask the objects, I would use the PyTorch library Detectron2 [3] from Facebook research to detect the class of objects I want to mask and compute their masks. Then I will change the size or number of holes progressively in order to observe at which point the model has difficulties. Since the model is sensitive to the color within the masked pixels, there is a choice on how to color the masked region. Possible colorings of the masked pixels can be either white or the average of N-pixels around the mask.

The above will be applied on different categories of images such as: landscapes, faces, architectures, urban environments, motifs, wildlife. The following "dataset" I will create from any of the current available datasets (Paris Street View, ImageNet, CelebA, …).

However Deep neural network in-painting usually suffers from lack of texture in the masked region, the creation of artificial edges around the hole and obvious color contrasts [2]. Hence a possible improvement would be to try and implement partial convolution [2] in the SinGAN model and observe their performance in comparison with other architectures (U-Net).

3. Evaluation

The results from SinGAN can be evaluated in a qualitative or quantitative way just like the paper:

1. Use the Single Image Fréchet Inception Distance (SIFID) score.
2. Compare performance with other SOTA architecture such as NST or the U-Net architecture.
3. Survey to other students where you ask if the image is real or fake.

References

[1] Tamar Rott Shaham, Tali Dekel, Tomer Michaeli. SinGAN: Learning a Generative Model from a Single Natural Image. (2019) *arXiv:1905.01164.*

[2] Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, Bryan Catanzaro, Image Inpainting for Irregular Holes Using Partial Convolution. NVIDIA Corporation. (2018) *arXiv:1804.07723*.

[3] Yuxin Wu and Alexander Kirillov and Francisco Massa and Wan-Yen Lo and Ross Girshick. Detectron2 (2019) *https://github.com/facebookresearch/detectron2*

[4] Leilei Cao, Tong Yang, Yixu Wang, Bo Yan, Yandong Guo.Generator Pyramid for High-Resolution Image Inpainting (2020). *arXiv:2012.02381v1*

[5] Jireh Jam, Connah Kendrick, Kevin Walker, Vincent Drouard, Jison Gee-Sern Hsu, Moi Hoon Yap. A comprehensive review of past and present image inpainting methods. *Computer Vision and Image Understanding. Volume 203.* (2021)

[6] Github code of the official paper [1] at *https://github.com/tamarott/SinGAN*

[7] Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B., Patchmatch: A randomized correspondence algorithm for structural image editing

[8] Levin, A., Zomet, A., Peleg S., Weiss, Y., Seamless image stitching in the gradient domain, *European Conference on Computer Vision.* (2004)

[9] Park, E., Yang, J., Yumer, E., Ceylan, D., Berg, A.C., Transformation-grounded image generation network for novel 3d view synthesis, *IEEE Conference on Computer Vision and Pattern Recognition.* (2017)

[10] Zhengwei Wang, Qi She, Tomas E. Ward. Generative Adversarial Networks in Computer Vision: A Survey and Taxonomy (2020). *arXiv:1906.01529*