CS230

Deep Layer Regeneration: Image Reconstruction

Jesus Mendez
Stanford University
Stanford, CA
mendezj@stanford.edu

Abstract

Technological advances have impacted our lives in ways we could not have anticipated; not only influencing the way we connect, but also the way we document our daily activities. In current society, many individuals are connected by the an array of social media platforms, and undergo substantial efforts to perfectly curate their online facade. Getting the perfect picture of an exotic place you visited, or snapping a loving memory with a loved one, has never been as important to share with the cyber world as it is now. However, despite our best efforts, these picturesque memories cannot always be captured the way we intend them to. Giant herds of people break the sublime scenery, bystanders walk in front of you, colluding your picture, to even focused sabotage by our friends can all ruin a perfect moment. Hence, I explore the idea of replacing unwanted people from images. This is done by using an end-to-end trained Faster R-CNN, which uses a ResNet-50-FPN backbone for people object detection, and a Generative Adversarial Network; creating a life like image.

1. Introduction

In today's society, we have grown a deep desire to have an online presence that we continuously groom in order to best illustrate who we are. Particularly this can be seen by the vast amount of pictures uploaded to social media outlets. Technology has allowed us to be more connected than ever before, while also enabling us to very easily document our everyday lives. This combination coupled with the ever growing mentality of "pictures or it didn't happen" fosters a further desire for users to take and share pictures of their daily lives. This creates a challenge of attempting to capture our environments structure, through its imperfections and chaos.

In order to get that perfect picture, we would have to remove objects from photo and replace them with realistic renderings. This is an interesting challenge to explore because it gives users exact control of how they want to remember a memory. Photos will no longer contain random bystanders, photobombs, and or people, ultimately giving users the ability to better express themselves through pictures.

This process begins by taking an image and classifying the various objects within it. Masks associated within the objects boundary box are modified with random noise. Thus disturbing its real-image pixel distribution and allowing Generative Adversarial Networks to iteratively construct a life-like image.

2. Related Work

Facebook AI Research (FAIR): Object Detection
During my investigation I came across dozen of groups who have resolved challenges regarding object detection. Of those, I selectively focused on the work done by Facebook AI Research team as they have pre-trained models and code available on their github account for rapid prototyping. These models were trained in a cluster of GPU's as training is very costly and therefore time consuming. Satisfactory level of performance are seen in these initial pretrained models, to the point they have been extended to not just people detection but, cars, bikes, trees etc…

*Generative Adversarial Network*s
Generative Adversarial Networks (GANs) have shown enormous promise regarding generating real natural images. Most recently, applications regarding generating human like renderings of imaginary celebrities [9]. Despite these impressive "natural" renderings, generated images still have room for improvement. Therefore one stage of the proposed method multi-step process will involve GANs ability to generate natural images through backgrounds which tend to have bold structure.

3. Dataset and Features

The dataset for this project was originally introduced by People in Photo Albums (PIPA) dataset, a large-scale recognition dataset collected from the social media

platform Flicker [7]. Dataset consist of 37,107 photos by 63,188 instances of 2,356 identities and examples are shown in **Figure 1A** below. Most importantly, this dataset will be used to train the Mask R-CNN. As training will be costly, we will apply transfer learning to obtain weights from a similar objective [6].

Figure 1A: People in Photo Albums (PIPA) dataset

The second dataset Places365 contains 2.5 million images across 205 scenes. Only a small fraction of about 600 modified images will be used to train the Generative Adversarial Network. Samples of the dataset can be found in **Figure 1B**.

Figure 1B: Places365

Pre-trained Models

Facebook's Dectron Baseline ran on a Basin Server with a cluster of 8 NVIDIA Tesla P100 GPU. Only horizontal flipping data augmentation was used [8]. The various backbone models were pre-trained on Imagine-Net-1k/5K dataset.

Case	Type	Train time total (Hr)	Inference time (s/im)
R-50-FPN	Mask R-CNN	44.9	0.099 + 0.018
R-101-FPN,	Mask R-CNN	49.7	0.126 + 0.017

Figure 2: Baselines with Batch Norm

Results from Facebook Detectron using the above pretrained model are shown in **Figure 3**.

Figure 3: Detectron Demo Output

4. Methods

The modified masked R-CNN is in essence the same conceptually as the Mask R-CNN. Except with the modification of injecting a non-transparent hue to the objects identified. The idea is that in a later stage, the user will oversee which objects want to be kept and or removed. The objects left identified with the solid colored hue, will result in disturbing the real image distribution of that picture.

The GANs on the other hand was trained with a different dataset whose sole responsibility was to restore missing pieces of its fed images. Therefor, the GANs would learn how to modify the solid colored patch and iteratively bring it closer to the real image until it passes [1].

Ultimately we have a Neural Network that is synchronously disjoined. In order for the GANs to work properly it must not know that the image patch that it modifies is not an object but believe it is completing a pattern. Thus, it believes it is completing a real image patch and brings it closer to the perceived real image. Full proposed neural network architecture is shown in **Figure 4.**

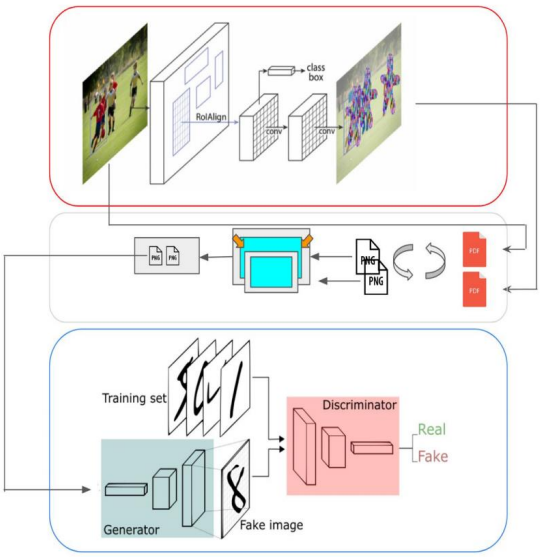

Figure 4: Neural Network Architecture

5. Results

The following image results, are broken into two sections respective of their models. Mask R-CNN and the GANs can be found in **Figure 5 & 6** respectively. The GANs network was trained on an 8 core Intel Xeon E5-2686 processor.

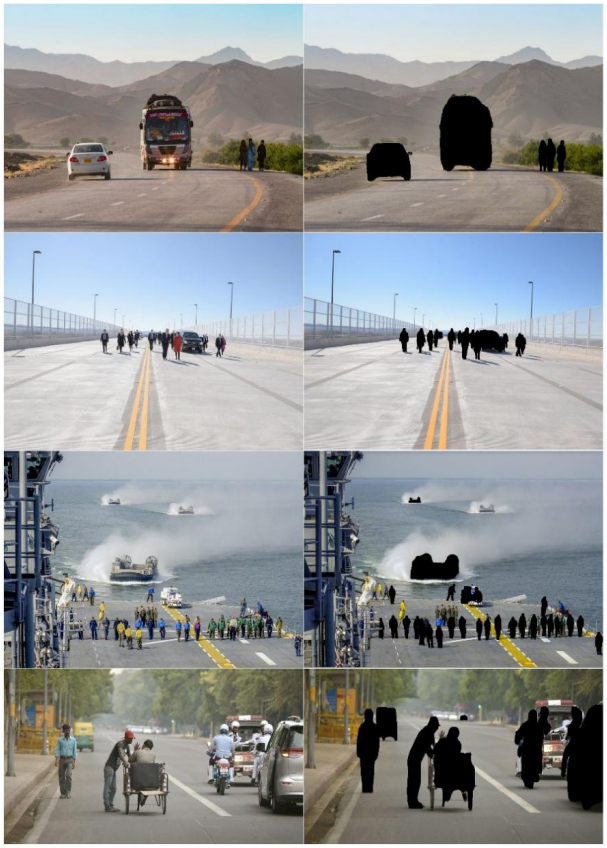

Figure 5: Mask R-CNN

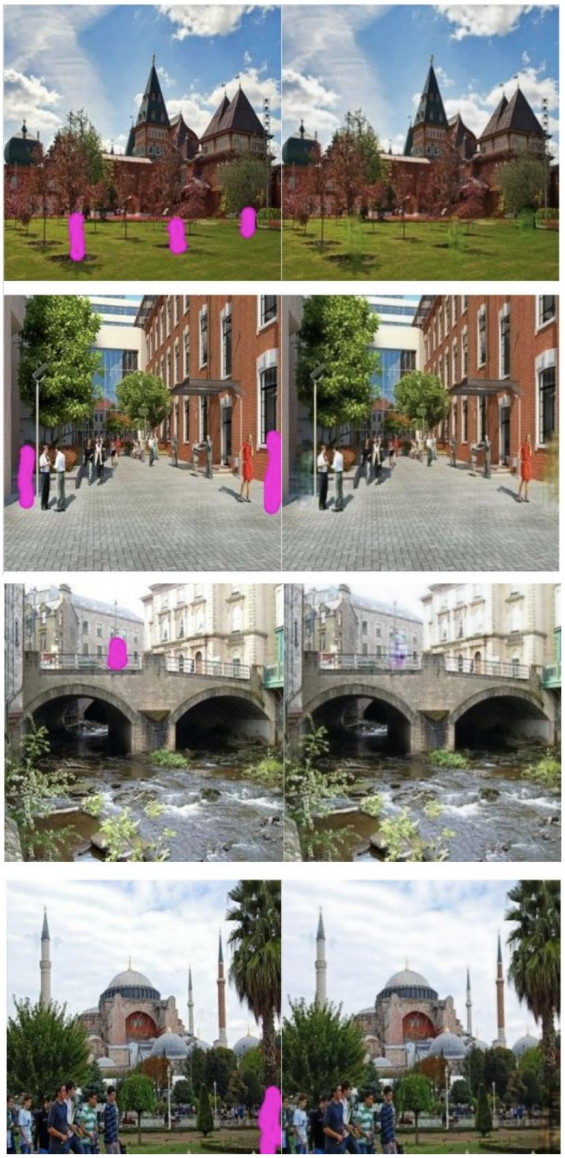

Figure 6: General Adversarial Networks

6. Discussion

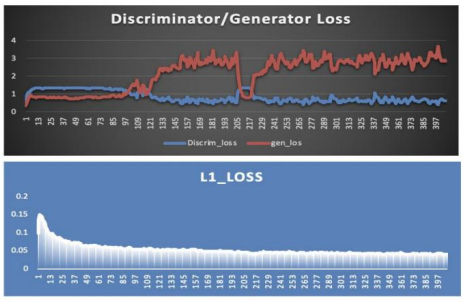

Figure 7: General Adversarial Network

Garbage image data was inserted into the dataset to mimic blurry and or poor quality images. Given the small dataset, the training data was heavily skewed towards light nature related environments.

Figure 8: Complete Neural Network Architecture: Elephant Chase

Figure 9: Complete Neural Network Architecture: Sample

7. Conclusion

Mask R-CNN and Generative Adversarial Network combination are effective in translating an image from one domain to the other. While the injected hue was specified, experimenting with other highly saturated colors provided similar results and is non-restrictive. Slight discoloration is caused by the skewed light nature environments that the dataset contained. Providing greater dataset scene variation and larger training set would offset these smaller discrepancy.

8. Future Work

My focus will be in automating training data to resolve unseen environments. Some of the better visually performing outputs were due to dataset coverage. In addition the results seen, were trained on a CPU with a max epochs of 50. Hyperparameter tuning and training for longer on a GPU will only increase the end results. I

am confident, that with these changes an undistinguishable image will arise.

9. Acknowledgements

I would like to thank the TA team for being not only a constant inspiration, but providing great assistance in the forums and presenting hands-on material during office hours. To Professor Andrew, thank you for your kindness in teaching, as the learning platform makes it easy to become excited and build rewarding projects.

10. Contributions

As an individual project this section may be omitted.

11. Code

https://github.com/MendezJesus/DeepLayerRegeneration

References

[1] Ian Goodfellow, Yoshua Bengio, Aaron Courville: Deep Learning, 2016

[2] Jianwei Yang, Anitha Kannan, Dhruv ZBatra, Devi Parikh. LR-GAN: Layered Recursive Generative Adversarial Networks for Image Generation. arXiv reprint arXiv:1703.01560

[3] Jianwei Yang, Anitha Kannan, Dhruv Zbatra, Devi Parkh. Pytorch code for Layered Recursive Generative Adversarial Networks https://github.com/jwyang/lr-gan.pytorch

[4] Facebook AI Research (FAIR) https://github.com/facebookresearch/Detectron

[5] US Government Works https://www.usa.gov/government-works

[6] Ning Zhang, Manohar Paluri, Yaniv Taigman, Rob Fergus, Lubomir Bourdev: Beyond Frontal Faces: Improving Person Recognition Using Multiple Cues: arXiv:1501.05707

[7] Ning Zhang, Manohar Paluri, Yaniv Taigman, Rob Fergus, Lubomir Bourdev: DATASET: Beyond Frontal Faces: Improving Person Recognition Using Multiple Cues: https://people.eecs.berkeley.edu/~nzhang/piper.html

[8] Detectron Model Zoo and Baseline https://github.com/facebookresearch/Detectron/blob/master/MODEL_ZOO.md#end-to-end-faster--mask-r-cnn-baselines

[9] Tero Karras, Timo Aila, Samuli Laine, Jaakko Lehtinen: Progressive Growing of GANs for Improved Quality, Stability, and Variation https://research.nvidia.com/sites/default/files/pubs/2017-10_Progressive-Growing-of/karras2018iclr-paper.pdf

[10] Kaiming He, Georgia Gkioxari, Piotr Dollar, Ross Girshick: Mask R-CNN arXiv:1703.06870

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros: Image-to-Image Translation with Conditional Adversarial Networks adXiv:1611.07004

[12] Image-to-Image Translation with Conditional Adversarial Nets: https://phillipi.github.io/pix2pix/