Improved Unsupervised Style Transfer - Mask CycleGAN

Jianlin Du
Carnegie Mellon University
Pittsburgh, PA 15213
jianlind@andrew.cmu.edu

Zehao Guan
Carnegie Mellon University
Pittsburgh, PA 15213
zehaog@andrew.cmu.edu

Yi Zhou
Carnegie Mellon University
Pittsburgh, PA 15213
yizhou3@andrew.cmu.edu

Wendi Cui
Carnegie Mellon University
Pittsburgh, PA 15213
wendic@andrew.cmu.edu

1 Introduction

CycleGAN has achieved great results in unsupervised style transfer. However, it is not precise in certain scenarios. For example, in figure 1, when the original CycleGAN transfers a horse to zebra, it applies stripes to not only the horse but also the human riding the horse. Besides, the background of the image is also patternized with the stripes and has lower saturation. To help remedy the above problem, we propose the Mask CycleGAN which consists of a segmentation network (here we use Mask R-CNN) that generates a mask of the target object, and a CycleGAN that takes a four-channel image (RGB+mask) as input.

Our Mask-CycleGAN achieves a satisfying performance by rendering precise images with patterns applied only on the target object. At the same time, the background is less noisy and more realistic now.

Figure 1: Failure of CycleGAN

2 Related Work

2.1 CycleGAN

Contrary to GAN with only one generator from input to the generated image, CycleGAN, proposed by Zhu et al. (2017)[1] adds a second generator which converts a generated image back. So there is a constraint as it needs to keep the difference between input and converted back image down.

However, CycleGAN has a less than state-of-art ability to identify target objects. For example, when trying to convert an image of horse to a zebra, stripes may be applied to the background and human rider. We proposed to add a mask to fix this problem.

2.2 Mask R-CNN

Mask R-CNN, proposed by He et al.(2017)[2] is a conceptually simple, flexible and general framework for object instance segmentation. The applied approach efficiently detects the objects in image while simultaneously generating a high-quality segmentation mask for each instance.

3 Methods

3.1 Pipeline

There are two important components as illustrated in figure 2. One is Mask R-CNN (or other segmentation networks) for generating mask of the target object; The other is a CycleGAN which takes images with four channels (RGB+mask). This pipeline is a meta methods in that the Mask R-CNN could be substituted with better segmentation network in the future.

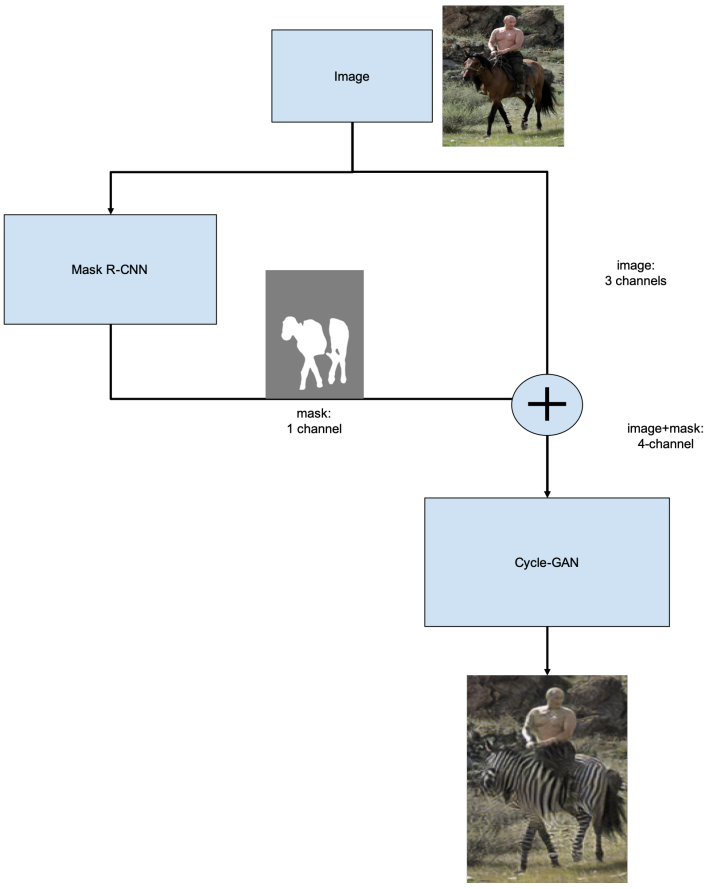

Figure 2: Model architecture

More specifically, for the mask, the region of target object is set to 255, while the background area is set to 127. At first, we tried mask of 255 and background of 0, but the generated output had a very dark background.

When doing style transfer, the picture is fed into Mask R-CNN to generate the mask. Then, RGB channels and the mask are stacked together and fed into CycleGAN.

3.2 CycleGAN Modification

Original CycleGAN would take three channels of R, G, B as the inputs of the model. We modified the generator of CycleGAN to take a fourth channel (mask) besides the RGB channels, so that our modified CycleGAN could take input of masks produced from the Mask R-CNN.

3.3 Mask R-CNN Fine Tuning

We started with a Mask R-CNN pre-trained on COCO dataset. COCO is a large-scale object detection, segmentation, and captioning dataset. It contains segmentation of 80 classes of common objects including people, horses, and zebras. In order to further improve the ability of the network to produce masks, we fine-tuned a pre-trained model to only recognize people, horses, and zebras and ignore other classes. Then, we preprocessed the dataset of horses and zebras provided in the CycleGAN repository with the fine-tuned Mask R-CNN to generate the masks.

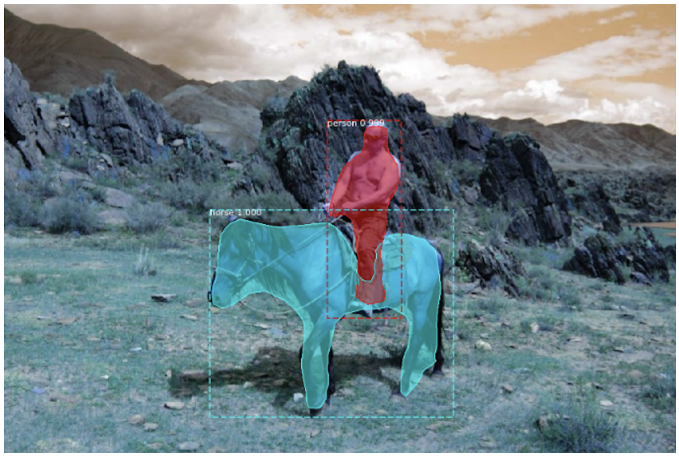

Figure 3: Mask R-CNN output mask

As shown in figure 3, Mask R-CNN model can well handle the task of generating mask. Here is an example output of the Mask R-CNN. We can see that the horse and the rider are generally well separated, and CycleGAN would have a much better idea on what to style transfer after getting the mask.

Then, we used Mask R-CNN to preprocess the training dataset of horses and zebras of CycleGAN. To ensure that they are on the same scale as the other three channels in CycleGAN, the masked area is represented as 255 and the background is represented as 127.

4 Datasets

We randomly selected 1500 images from COCO (Common Object in Context) with categories person, horse, and zebra, respectively for the whole experiments and 200 images for testing.

Contrary to the Mask R-CNN, for our own model we would only concentrate on horse and zebra, regardless of people. And we utilize the database shown below for our training and testing process.

For CycleGAN, we included 1068 horses and 1335 zebras in the training set and 141 images of each kind in the testing set.

Table 1: Training Dataset of Mask R-CNN and Mask CycleGAN

	Train	Test
horse	1500	200
zebra	1500	200
person	1500	200

	Train	Test
horse	1068	141
zebra	1335	141

5 Results

From left to right: input, output of CycleGAN and output of Mask CycleGAN.

- **Horse to zebra**

- **Zebra to horse**





Above are some results of images (from testing set) being fed into original CycleGAN and our Mask CycleGAN. Our model output better quality images and successfully distinguished the horse from the rider.

One problem we face is that the segmentation network may fail to recognize all target objects in the image. However, the above mistake also proves that CycleGAN has fully perceived the meaning of the mask channel, and it reveals to us the functionality of applying pattern to any area that you want by picking a particular mask or manually crafting a mask.

- **Dog to zebra**


6 Conclusion

Our model successfully solve original CycleGAN's problem of sometimes being confused target object with background, and proved that Generator of CycleGAN can successfully perceive the meaning of masks. In addition, our pipeline produced better quality images with less noise in background.

A limitation of our approach is that although current segmentation networks are good in most circumstances, they may still make mistake, which may affect the result. However, we believe there will be better and more reliable segmentation techniques to produce more accurate mask in the future.

One of the possible future improvements is to use a smooth/fuzzy mask, instead of setting pixel values to either 255 or 127. Meanwhile, it may be worth trying other value combinations for mask/background, e.g. positive for the target area and negative for background area.

7 References

[1] Zhu, J. Y., Park, T., Isola, P., Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision (pp. 2223-2232).

[2] He, K., Gkioxari, G., Dollár, P., Girshick, R. (2017). Mask r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 2961-2969).

[3] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). Generative adversarial nets. In Advances in neural information processing systems (pp. 2672-2680).

[4] Einstein, A., B. Podolsky, and N. Rosen, 1935, "Can quantum-mechanical description of physical reality be considered complete?", Phys. Rev. 47, 777-780.

[5] Gatys, L. A., Bethge, M., Hertzmann, A., Shechtman, E. (2016). Preserving color in neural artistic style transfer. arXiv preprint arXiv:1606.05897.

[6] Wang, H., Liang, X., Zhang, H., Yeung, D. Y., Xing, E. P. (2017). Zm-net: Real-time zero-shot image manipulation network. arXiv preprint arXiv:1703.07255.

[7] Jing, Y., Liu, Y., Yang, Y., Feng, Z., Yu, Y., Tao, D., Song, M. (2018). Stroke controllable fast style transfer with adaptive receptive fields. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 238-254).

[8] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. In European conference on computer vision (pp. 740-755). Springer, Cham.

8 Appendix

GitHub link: https://github.com/Yi-Zhou/dl_proj

Poster link: https://github.com/Yi-Zhou/dl_proj/blob/master/poster_785.pdf