IMAGE INPAINTING and OBJECT REMOVAL with DEEP CONVOLUTIONAL GAN

Qiwen Fu, Yuxin Yang, You Guan
{qiwenfu, yuxiny, you17} @ stanford.edu

PROBLEM DEFINITION

Remove unwanted objects from pictures.

When you take a picture of a beautiful view of Tahoe Mountains, suddenly you realize you have some random person keeps annoyingly appearing in the carefully taken picture (Figure 1a). Or when you want to take a picture of a snowy day in Lake Tahoe, but it is too crowded with skiers to take a good picture (Figure 1b). Or some stranger appearing on your picture of Death Valley (Figure 1c).

Figure 1. (a) Unwanted character in Tahoe Mountain. (b) Unwanted skiing crowds in picture. (c) Random tourists on Death Valley.

Draw a box, we remove the unwanted for you:

Figure 2. Unwanted objects manually blackboxed by users.

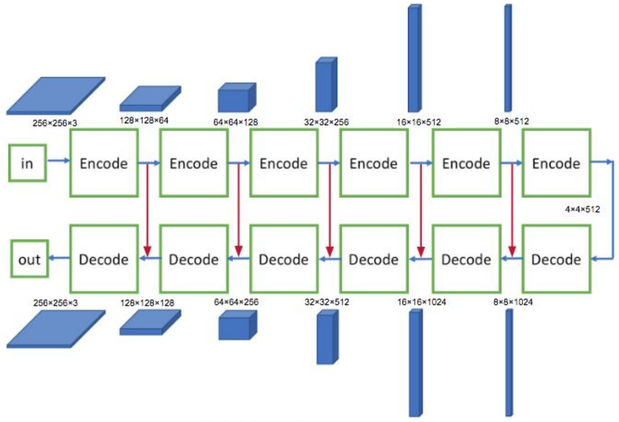
Figure 3. Our picture inpainting results.

DATA SET and PRE-PROCESSING

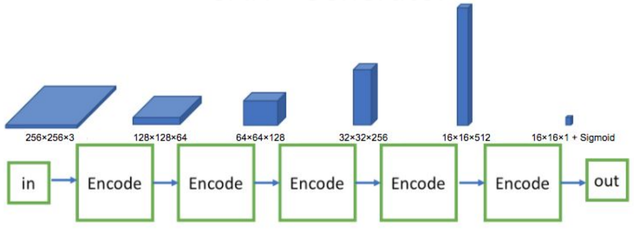
MIT Places: 250 Million 256×256 pictures under different categories, about 10K pictures under each category. We used mountain, snowfield, and pasture for training. After manually removing pictures with people or sharp objects, we were left with 15000 pictures for training.

CONDITIONAL DC-GAN

NETWORK ARCHITECTURES



GAN - Generator



GAN - Discriminator

FUTURE WORK

- Fine-tune the model to make the output image less artificial and more realistic
- Train on a compound dataset to get a general model
- Develop mobile apps for image inpainting and object removal applications
- Integrate with some traditional image inpainting methods to get better results

MORE RESULTS

Figure 4: First row shows the original picture before processing. Second row shows the blackboxed objects we want removed and impainted. Third row shows our results produced by DC-GAN.

CHALLENGES

- DCGAN model is difficult to train. Both discriminator and generator has to be tuned such that they are learning at the same pace.
- We had to manually retrain Discriminator from scratch several times throughout the training, or it simply becomes too good.
- The GAN model takes around 50 minutes to train 1 epoch, and it usually takes ~40 epochs in order to get acceptable results.
- Memory overhead is high. CUDA crashes from time to time, leading to wasted training efforts.

CONCLUSIONS

In our project, we successfully demonstrated the power of DC-GAN in picture inpainting. We can further conclude that GAN has a huge potential in the future of image processing applications, especially in fields related to image restoration and recovery.