"Deep Fakes" using Generative Adversarial Networks (GAN)

Tianxiang Shen, Ruixian Liu, Ju Bai, Zheng Li
{tis038, rul188, jub010, zhl153} @ucsd.edu

GOAL

The topic is inspired by some useful applications of GAN to assist designing works. We use a cycleGAN network to obtain "fake" images from some input real images, namely, transfer a hand bag to a backpack and vice versa.

A demonstration of out result

DATA

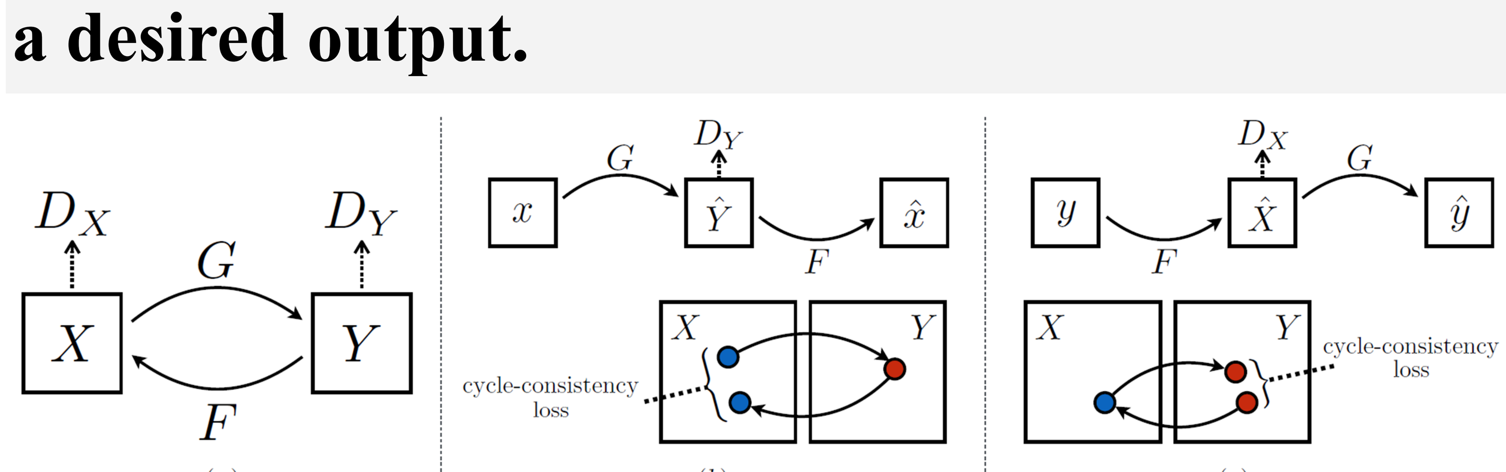
The data we used to train our cycleGAN network are downloaded from Google Images, in total 950 images for backpacks and 760 for handbags. They are colored but not normalized, we normalized them in program.

FEATURES

GANs are unsupervised learning algorithms. The cycleGAN network we used can automatically learn features combining many aspects properly such as colors, lines and corners of input images.

MODELS

We use a cycleGAN network to generate "Deep Fakes". CycleGAN combines two GAN networks together and use 2 loss functions: adversarial loss to ensure the ability to generate reasonable fake images and cycle-consistency loss to guarantee a individual input is mapped to a desired output.



(a) Our model contains two mapping $G : X \to Y$ and $F : Y \to X$, and associated adversarial discriminators D_Y and D_X. D_Y encourages G to translate X into outputs indistinguishable from domain Y, and vice versa for D_X and F. To further regularize the mapping, we introduce two cycle consistency losses that capture the intuition that if we translate from one domain to the other and back again we should arrive at where we started.

(b) Forward cycle-consistency loss: $x \to G(x) \to F(G(x)) \approx x$

(c) Backward cycle-consistency loss: $y \to F(y) \to G(F(y)) \approx y$

◆**One adversarial loss:**

$$L_{GAN}(G, D_Y, X, Y) = E_{y\sim p_{data}(y)}[\log D_Y(y)] + E_{x\sim p_{data}(x)}[\log(1 - D_Y(G(x))]$$

◆**Cycle-consistency loss:**

$$L_{cyc}(G, F) = E_{x\sim p_{data}(x)}[||F(G(x)) - x||_1] + E_{y\sim p_{data}(y)}[||G(F(y)) - y||_1]$$

◆**Total loss:**

$$L(G, F, D_X, D_Y) = L_{GAN}(G, D_Y, X, Y) + L_{GAN}(F, D_X, Y, X) + \lambda L_{cyc}(G, F)$$

RESULTS

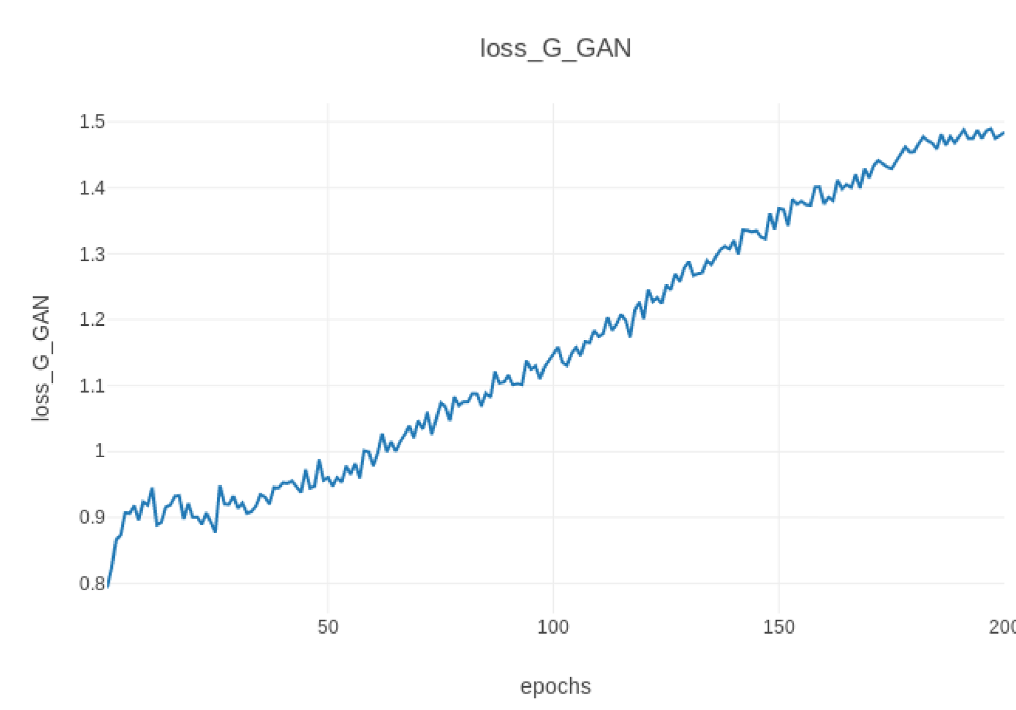
◆Handbag: 660 for training and 100 for testing

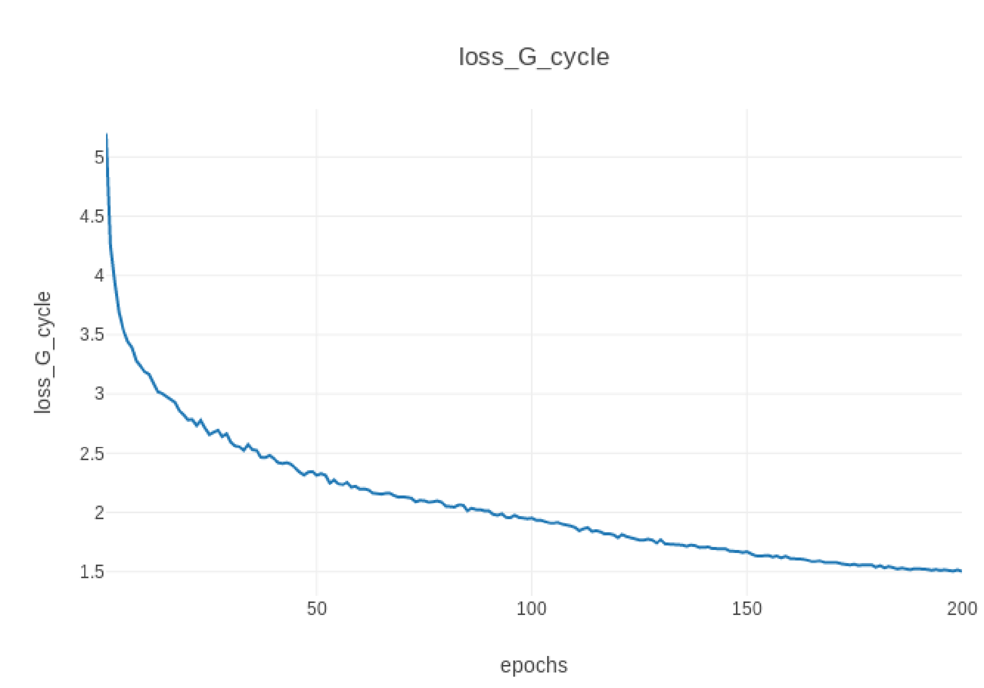
◆Backpack: 801 for training and 149 for testing

Error:

$$loss\_G\_GAN = E_{x\sim p_{data}(x)}\left(D_y(G(x)) - 1\right)^2 + E_{y\sim p_{data}(y)}\left(D_x(F(y)) - 1\right)^2$$

$$loss\_G\_cycle = L_{cyc}(G, F)$$





DISCUSSION

We used cycleGAN to transfer handbags to backpacks and vise versa. The results we got are relatively satisfying, with no obvious unreasonable distortions appearing. This is because the introduced cycle-consistency loss can keep the transfer process running in a reasonable manner.

POTENTIAL DIRECTIONS

◆Applying batch processing, and downsize the output image to speed up the training process.

◆Use the data sets on UC Berkeley's repository to training our model. Implement transform different object transfiguration, style transfer.

References:
[1] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks," 2017 IEEE International Conference on Computer Vision (ICCV), 2017.
[2]M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised Image-to-Image Translation Networks," Advances in Neural Information Processing Systems (NIPS), 2017.
[3]L. A. Gatys, A. S. Ecker, and M. Bethge, "Image Style Transfer Using Convolutional Neural Networks," 2016.