Image-to-Image Translation with Conditional Adversarial Netwoks

P. Arvanitis - N. Sourlos - P. Talluri

Deep Learning Course - Reproduction and Research

Faculty of Electrical Engineering, Mathematics and Computer Science - TUDelft

Instructor: J.v.Gemert   Assistant: P. Rajput

Learn more here

Delft University of Technology

Abstract

- Generative Adversarial Networks (GANs) are relatively new in the field of deep learning. Training on a particular data distribution, they are able to generate new samples which belong to the same distribution.
- Conditional GANs (cGANs) take as input samples from one distribution and generate samples of another distribution. These networks are conditional on the input, meaning that they rely on both the input and noise to generate an output.
- Image-to-Image translation makes use of the cGANs to map input to output images. In pix2pix, the Discriminator penalizes structure in an image by enforcing a joint loss function of a number of pixels, using PatchGAN.
- The results of image-to-image translation were evaluated using FCN-8s Semantic Segmentation. Table 1 of the pix2pix paper is reproduced for the label2photo experiment on the Cityscapes dataset, where we vary the objective function.
- We believe that the FCN criterion is insufficient in quantifying the quality of the generated images, particularly in cases where realism is valued.
- A new dataset with Asian faces and their sketches was used. *Face* → *Sketch* gave really good results while *Sketch* → *Face* failed

Diving into the objective function in cGANs

- The cGANs have the following loss function:

$$L_{cGAN} = E_{x,y}[log D_{\theta_D}(x, y)] + E_{x,z}[log(1 - D_{\theta_D}(x, G_{\theta_G}(x, z)))].$$

This loss function enforces crispness in the resulting image.
- The L1 loss is also used to obtain more accurate results in the low frequencies. The final objective is:

$$G* = arg \min_G \max_D L_{cGAN}(G, D) + \lambda L_{L1}(G),$$

where G and D are the Generator and Discriminator, respectively.
- The conditioning of the input on the Discriminator was also deemed important based on results.

Loss	Per-pixel acc. (ours)	Per-pixel acc.	Per-class acc. (ours)	Per-class acc.	Class IoU (ours)	Class IoU
L1	0.70	0.42	0.24	0.15	0.18	0.11
GAN	0.25	0.22	0.08	0.05	0.03	0.01
cGAN	0.59	0.57	0.20	0.22	0.14	0.16
L1 + GAN	0.65	0.64	0.20	0.20	0.15	0.15
L1 + cGAN	0.75	0.66	0.24	0.23	0.19	0.17

Table: FCN scores comparison for different objectives for the Cityscapes label → photo experiment

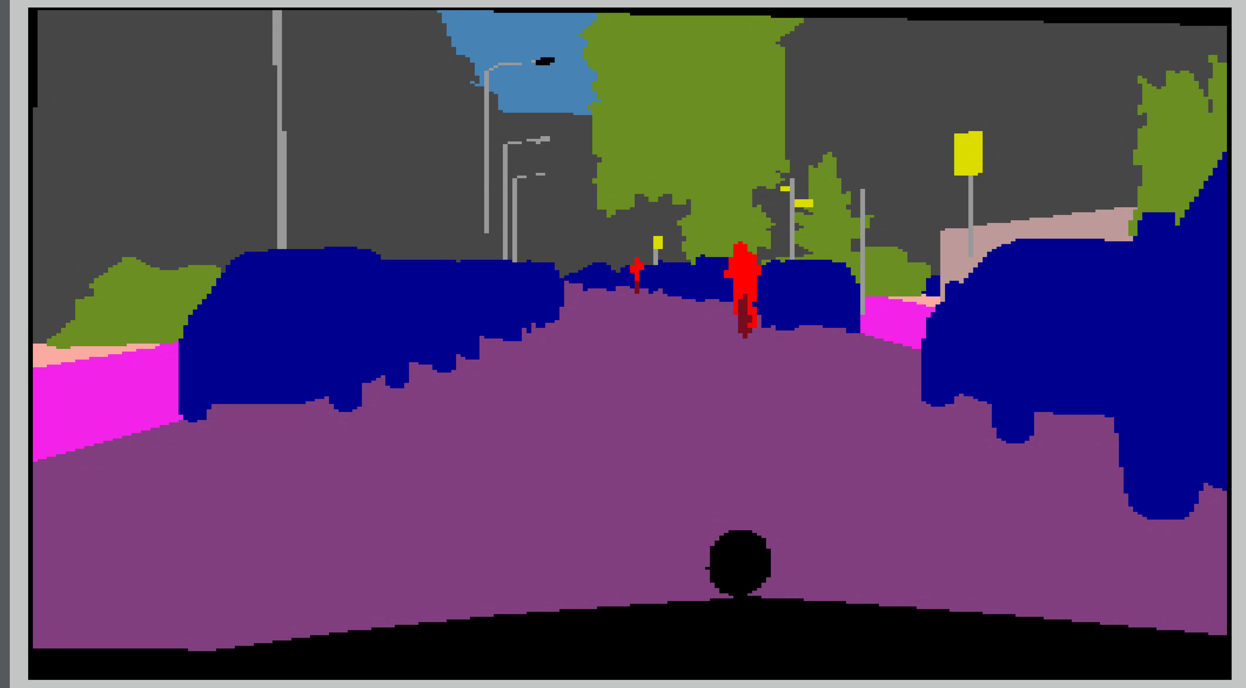
Face ↔ sketch

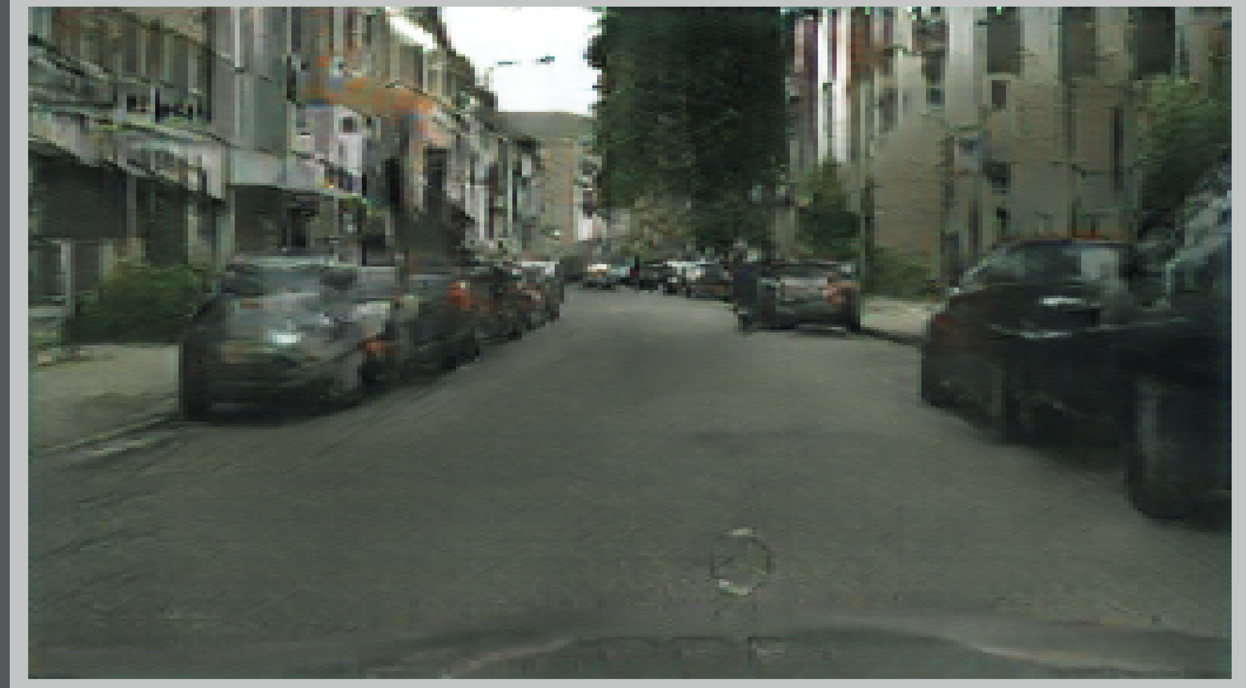
- A new dataset that includes 188 faces and their respective sketches from the Chinese University of Hong Kong (CUHK) student database was used.
- Two different types of cGANs were used, pix2pix and cycleGAN
- In pix2pix, images have to be in pairs while in cycleGAN images can be unpaired
- Different preprocessing of images required for each model
- For cycleGAN training set was chosen to contain 178 images, test set 10 whereas pix2pix training set size consists of 170 images, validation set of 8 and test set of 10 images.
- *Face* → *Sketch* works pretty well in both model types.
- *Sketch* → *Face* totally fails in cycleGAN while in pix2pix it produces a few realistic images. The distribution of faces is harder to learn, since it has a higher entropy

Hyperparameters Search

- A learning rate 3 times higher than the original one (0.006) leaves performance intact
- A learning rate 3 times less than the original one slightly decreases performance
- Reduction of total epochs from 200 to 100 in the case of the highest learning rate only reduces quality of results slightly
- A larger PatchGAN in the Discriminator (from 70 to 142) worsens results for the Cityscapes label → photo experiment

Cityscapes label → *photo*





New dataset face → *sketch*

References

- Dataset
- Tutorial
- CycleGANS and Pix2Pix
- A Gentle Introduction to Pix2Pix GANs
- Image-to-Image Translation with cGANs
- Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks
- Image Detection, Recognition and Classification
- Semantic Segmentation with Neural Networks
- Beginner's guide to GANs
- Example of a blog post on GitHub