Directed Studies Research Report

Zikun Chen, 55903421
zikun.chen@alumni.ubc.ca

1 Objective

This project aims to address the domain adaptation problem for semantic segmentation. In particular, we want to develop a method that closes the domain gap between the source domain(GTA5) and target domain(CITYSCAPES) by mapping data from the source domain to the target, and train a classifier using the source data stylized as the target.

2 Datasets

Cityscapes [2] is a real-world dataset consisting of 3475 high quality pixel-level annotated frames from 50 different cities. I used 2975 labelled images for training, and 500 for evaluation.

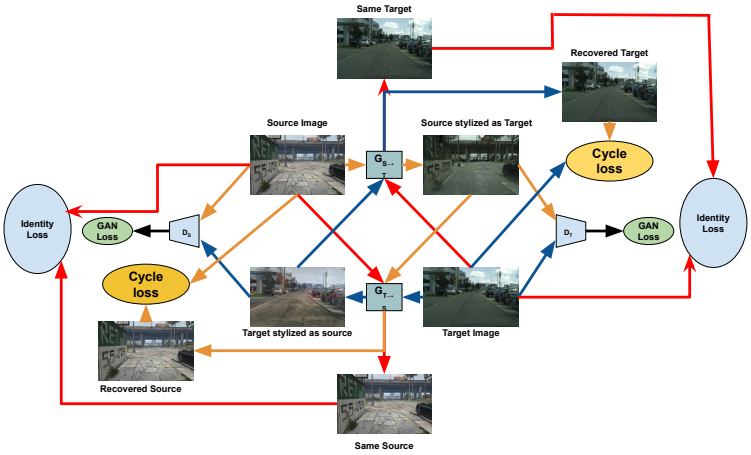
Figure 1: Sample pair of images from the CITYSCAPES data set.

GTA5 [5] is a synthetic dataset consisting of 24966 photo-realistic synthetic images with precise pixel-level semantic annotations. I went through the dataset and chose 16786 labelled images for training.

Figure 2: Sample pair of images from the GTA5 data set. As can be noticed, the car is driving on a sidewalk, and scenes like this are very common in the GTA5 dataset. Worried that this might lead to a significant domain gap unresolvable by aligning styles, I went through the dataset and excluded them from our training set.

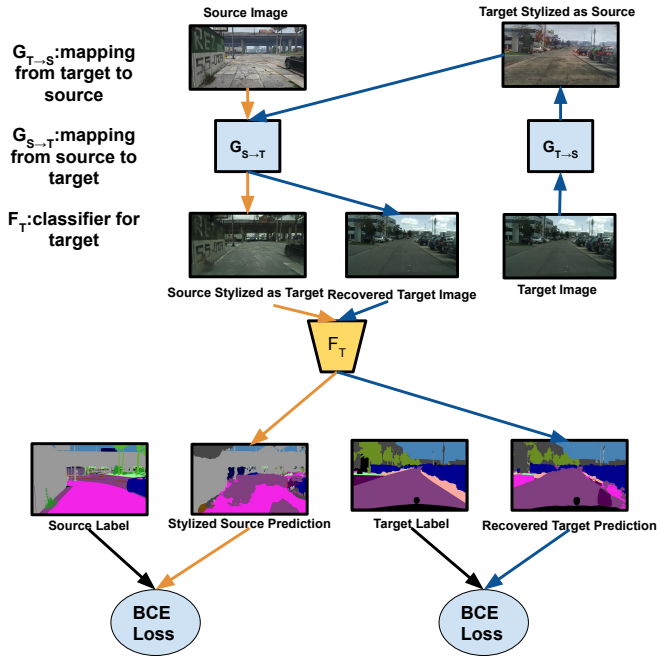
3 Models

3.1 Basic CycleGAN[7]

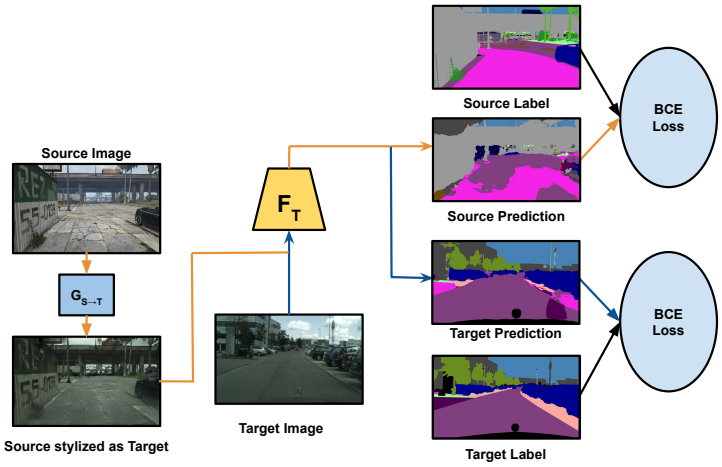


I added a classifier to preserve contents(cars, buildings) during domain transfer

3.2 Backprop from classifier to generators



3.3 Training Classifier



4 Notations

X_S : Input from the source domain (GTA5).
X_{SasT} : Input stylized as the target style (GTA5). This notation is introduced when the parameters of the $G_{S\to T}$: generators are not involved in back propogation.
Y_S : Label from the source domain.
X_T : Input from the target domain (CITYSCAPES).
Y_T : Input from the target domain.
F_T : Classifier for the target domain.
$G_{S\to T}$: Mapping from the source domain to the target.
$G_{T\to S}$: Mapping from the target domain to the source.
D_S : Discriminator for the source domain.
D_T : Discriminator for the target domain.
$CE(X, Y, F_T) := -\mathcal{L}_{(x,y)\in(X,Y)}[\sum_{k=1}^{K} \mathbb{1}_{[y=k]} \log[\sigma(F_T^k(x))]]$ cross entropy loss, X and Y are the input and the label, σ is the softmax function.

5 Loss Functions

The model learns mappings from the source domain to the target and the target to the source simultaneously. It is trained to produce samples that fools the discriminators in both domains.
$\mathcal{L}_{GAN}(G_{S\to T}, X_S, D_S, D_T) = \mathbb{E}_{x_s\in X_S}[\log D_S(x_s) + \log[1 - D_T(G_{S\to T}(x_s))]]$
$\mathcal{L}_{GAN}(G_{T\to S}, X_T, D_S, D_T) = \mathbb{E}_{x_t\in X_T}[\log D_T(x_t) + \log[1 - D_S(G_{T\to S}(x_t))]]$

The model learns to preserve contents through the cycle consistency loss and segmentation loss.
$\mathcal{L}_{Cyc}(G_{S\to T}, G_{T\to S}, X_S, X_T) = \mathbb{E}_{x_s\in X_S, x_t\in X_T}[\|G_{T\to S}(G_{S\to T}(x_s)) - x_s\|_1 + \|G_{S\to T}(G_{T\to S}(x_t)) - x_t\|_1]$
$\mathcal{L}_{G_{sem}}(G_{S\to T}, X_S, F_T, Y_S) = \mathbb{E}_{(x_s,y_s)\in(X_S,Y_S)}[CE(F_T(G_{S\to T}(x_s)), y_s)]$
$\mathcal{L}_{G_{sem}}(G_{S\to T}, G_{T\to S}, X_T, F_T, Y_T) = \mathbb{E}_{(x_t,y_t)\in(X_T,Y_T)}[CE(F_T(G_{S\to T}(G_{T\to S}(x_t))), y_t)]$

The classifier is trained on paired target data as well as paired source data stylized as target.
$\mathcal{L}_{sem}(X_{SasT}, X_T, Y_S, Y_T, F_T) = \mathbb{E}_{(x_{sast},y_s)\in(X_{SasT},Y_S),(x_t,y_t)\in(X_T,Y_T)}[CE(F_T(x_{sast}), y_s) + CE(F_T(x_t), y_t)]$

The overall cycleGAN training loss is:

$$\mathcal{L}_{cycleGAN}(X_S, Y_S, X_T, Y_T, F_T, G_{S\to T}, G_{T\to S}, D_S, D_T) = \mathcal{L}_{GAN}(G_{S\to T}, X_S, D_S, D_T) + \mathcal{L}_{GAN}(G_{T\to S}, X_T, D_S, D_T) + \mathcal{L}_{Cyc}(G_{S\to T}, G_{T\to S}, X_S, X_T) + \mathcal{L}_{G_{sem}}(G_{S\to T}, X_S, F_T, Y_S) + \mathcal{L}_{G_{sem}}(G_{S\to T}, G_{T\to S}, X_T, F_T, Y_T)$$

Which corresponds to solving the optimization problem

$$\underset{G_{S\to T}, G_{T\to S}}{\operatorname{argmin}} \max_{D_S, D_T} \mathcal{L}_{cycleGAN}(X_S, Y_S, X_T, Y_T, F_T, G_{S\to T}, G_{T\to S}, D_S, D_T)$$

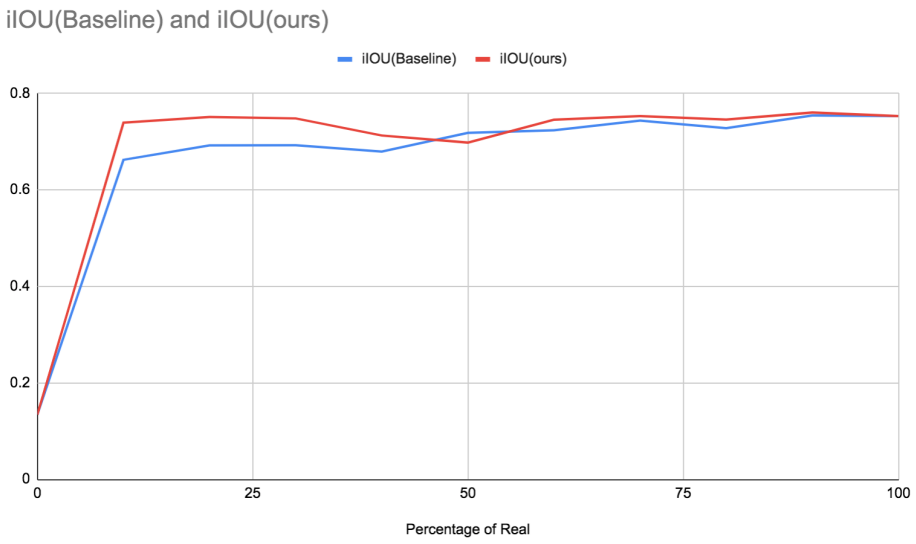
The segmentation network is updated separately with:

$$\underset{F_T}{\operatorname{argmin}} \mathcal{L}_{sem}(X_{SasT}, X_T, Y_S, Y_T, F_T)$$

6 Experiments

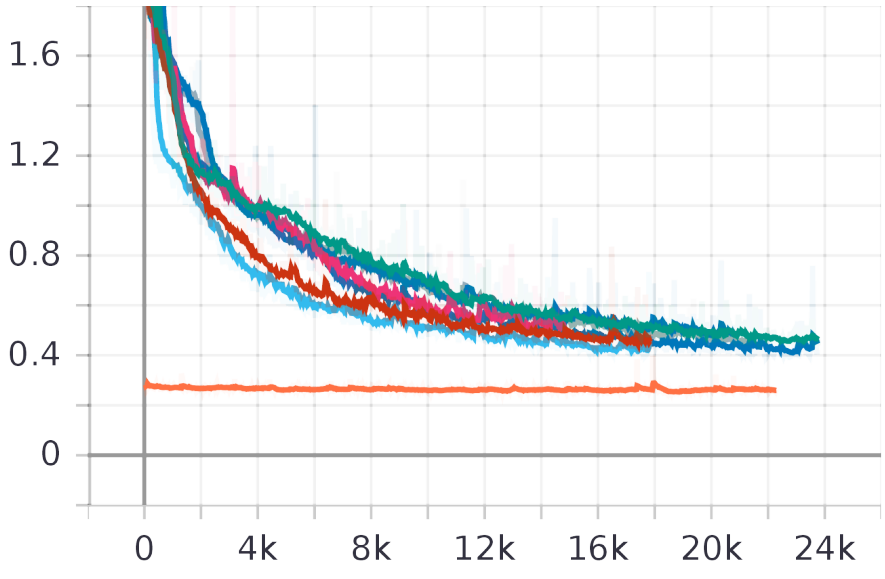
6.1 Changing percentages of real(target) data used in our model

In this experiment, we kept the dataset size fixed, and only included paired data from both domains, changing the percentage each takes.
When the percentage of real data is 0%, our classifier is only trained on data from the source domain(GTA5), and tested on CITYSCAPES validation set.
When the percentage of real data is 100%, our classifier is only trained on data from the target domain(CITYSCAPES), and tested on CITYSCAPES validation set.
When the percentage is between 0 and 100%, our classifier is trained end to end with CycleGAN that stylizes data from the source domain as the target.
The baseline is a classifier with the same architecture as ours, trained without style transfer.



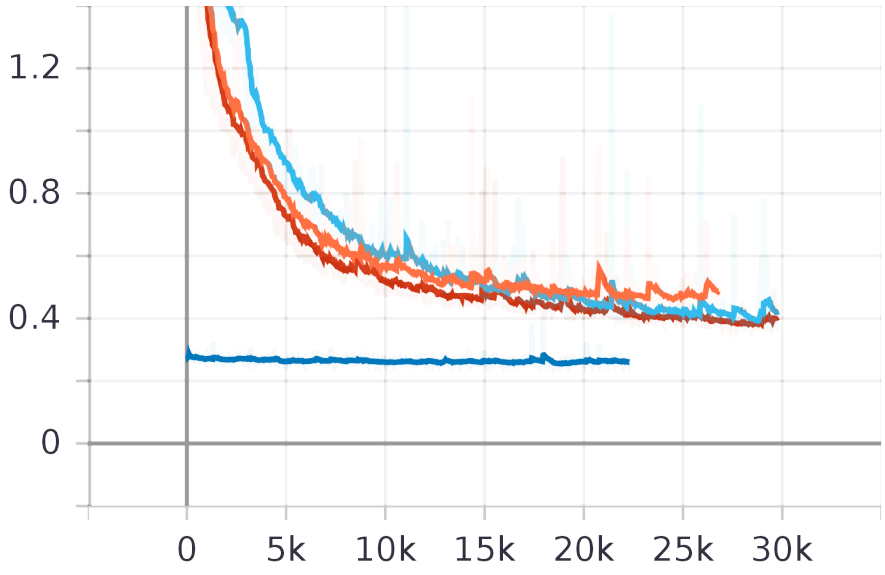
The evaluation metric used is weighted IOU. Our method achieved 75% accuracy with 10% of paired data from the target domain, but stops further improving as the percentage increases. Notice that the baseline reached %75 accuracy when around

90% of paired real data is used. But it's possible that the architecture for the classifier is too simple(U-Net) that it is not capable of learning more.



(1).png

Figure 3: keeping the training dataset size fixed, varying percentage of paired data (10% to 90%) in CITYSCAPES + GTA5 stylized as CITYSCAPES to train the segmentation network. The orange line is the fully supervised model using only CITYSCAPES pairs (oracle). It seems like style transfer might have negative effects.



(1).png

Figure 4: using all the paired data in CITYSCAPES + varying percentage of GTA5 (10%, 20%, 50% of total) stylized as CITYSCAPES paired data to train the segmentation network. The blue line is the fully supervised model using only CITYSCAPES pairs (oracle) It seems like style transfer might have negative effects.

new approach

In the first stage, train a segmentation network with GTA5 pairs to use as auxiliary loss for cycleGAN training. This should be replaced by some pre-trained network, i.e. a VGG.
In the second stage, train the normal cycleGAN, obtain a generator G that maps from GTA5 to CITYSCAPES and a discriminator D_T that tells if an image is from the target domain .
In the third stage, train a segmentation network F_T using stylized GTA5 pairs.
In the fourth stage, train F_T end to end with G, discriminator D_T and D_F. D_T tells if an image is from the target domain, in our case CITYSCAPES. D_F tells if a semantic map looks real, and we hope that adversarial training with D_F will force F_T to output realistic semantic maps for unsupervised domain CITYSCAPES.

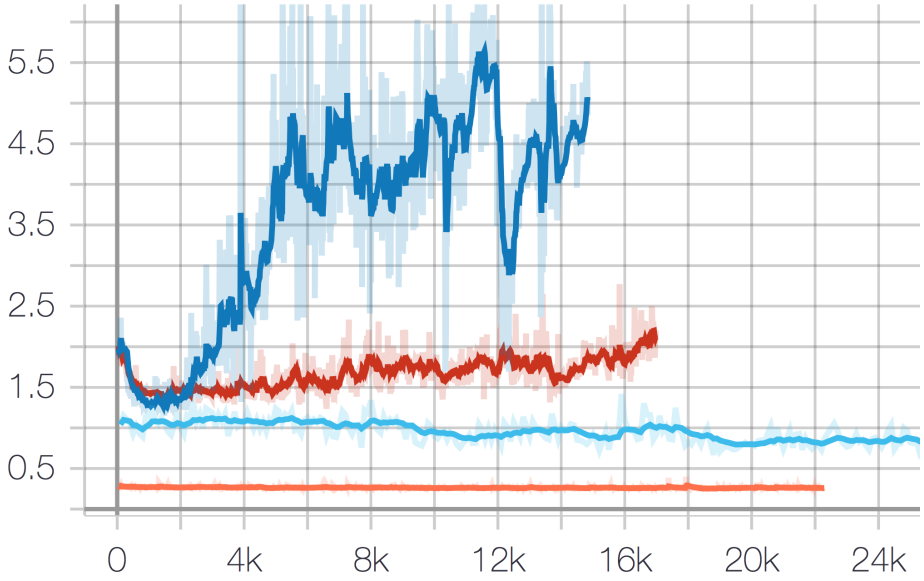


Figure 5: validation error of different models. newly proposed model- sky blue, source only- dark blue, stylized GTA5- red, and orange is the oracle model using all CITYSCAPES pairs.

TODOs

change baseline to deeplab v3 + resnet and retrain
train with current size cityscapes with full GTA5 and evaluate the performance.
train with full size cityscapes with cleaned GTA5 and evaluate the performance.
train with full size cityscapes with full GTA5 and evaluate the performance.
train Cycada [3] and compare
train Fcns in the wild [4] and compare
train [6] and compare
train [1] and compare

Issues

Adjust the weights on loss for segmenting stylized images when increasing the percentage of source data used.
The sample is unbalanced(most instances are road and sky), adjust the weights for different class?
training segmentation network and generators end to end might not work? keep segmen network fixed, train cygan, cygan converges and fixed, train segmen?

References

[1] Yuhua Chen, Wen Li, and Luc Van Gool. Road: Reality oriented adaptation for semantic segmentation of urban scenes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[2] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

[3] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213*, 2017.

[4] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv preprint arXiv:1612.02649*, 2016.

[5] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *European conference on computer vision*, pages 102–118. Springer, 2016.

[6] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7472–7481, 2018.

[7] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.