A Comparison of Synthetic-to-Real Domain Adaptation Techniques

Peter Trost

Eberhard Karls Universität Tübingen
Mathematisch-Naturwissenschaftliche Fakultät
Wilhelm-Schickard-Institut für Informatik
Lernbasierte Computer Vision

peter.trost@student.uni-tuebingen.de

October 1, 2019

Overview

Overview

thesis goals:

1. adapt images from synthetic to real domain using three different techniques
2. compare adaptation performance of these techniques

Overview

Autonomous driving

- use Deep Neural Networks
- need lots of data to perform well
- supervised learning yields best models
- labeling may be inaccurate (human annotators)
- data acquisition expensive

Overview

Domain Adaptation

Figure: example images of the two domains relevant for this thesis. Real domain (left, Cityscapes dataset) and synthetic domain (right, GTA5 dataset)

- may have a lot of data in one domain but not as much in another related one
- train machine learning model on the first domain
- perform domain adaptation: transfer that model to the other domain

Generative Adversarial Networks

- discriminator learns data distribution from training data (real)
- generator generates images (fake)
- GANs implement two player game:
  generator tries to fool the discriminator into labeling generated images as real

Generative Adversarial Networks

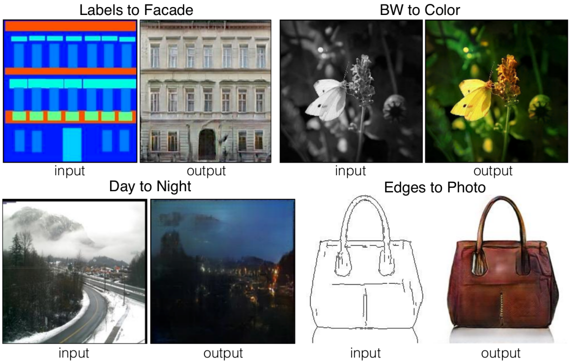


Figure: Examples of conditional inputs and generated images (output) of conditional GANs as described in [add citation here]

Overview

Discrepancy based

- **class criterion:** uses class label information
- **statistic criterion:** align statistical distribution shift between source and target domains
- **architecture criterion:** modify the network so it can learn more transferable features
- **geometric criterion:** bridges domains using their geometrical properties

Adversarial based

- **generative models:** GANs
- **non-generative models:** feature extractor learns to distinguish between source and target domain, maps features to common space.

Reconstruction based

- **encoder-decoder reconstruction:** combine encoder network for representation learning (features like edges, circle, etc.) and decoder network for data reconstruction
- **adversarial reconstruction:** use reconstruction error to make sure mapping from source to target and back to source results in the original datapoint

Overview

Why GANs?

- can generate images
- easier for humans to evaluate
- unsupervised learning

Overview

CycleGAN

- cycle-consistency loss: difference between source image and image translated to and then back from the target domain [add explanation graphic here]

CyCADA

- additional semantic loss:
  1. perform semantic segmentation on source image
  2. translate image to target domain
  3. perform semantic segmentation on that image

  difference between those two semantic maps describes the semantic loss [add visualization here]

SG-GAN

- additional gradient-sensitive objective: emphasize semantic boundaries by rendering distinct color/texture for each region [add visualization here]

Overview

Comparison Benchmark: Semantic Segmentation

- task of segmenting an image into multiple sets of pixels (superpixels)
- goal: change representation of image in order to make it more meaningful or simpler to analyze

Synthetic dataset: GTA5

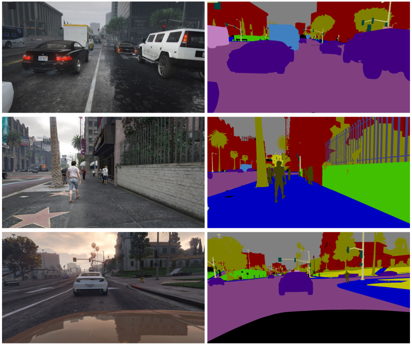


Figure: GTA5 images (left) with corresponding annotations (right)

- $\sim$ 25000 images
- obtained from Grand Theft Auto V

Real dataset: Cityscapes

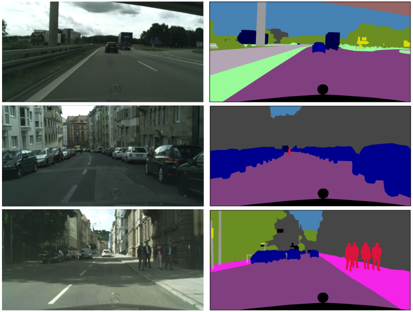


Figure: Cityscapes images (left) and corresponding fine annotations (right)

- 5000 images with fine annotations
- 20000 images with coarse annotations

Image Translation

- sampleset of 500 images randomly chosen
- images translated using CycleGAN, CyCADA, SG-GAN

Semantic Segmentation (DeepLabV3)

- DeepLabV3 performs semantic segmentation on translated images
- i.e. predicts label for every pixel in the image

Evaluation

- using the Cityscapes provided code
- Intersection over Union

$$\frac{\text{predicted pixels} \cap \text{ground truth pixels}}{\text{predicted pixels} \cup \text{ground truth pixels}}$$

Overview

Quantitative

category scores

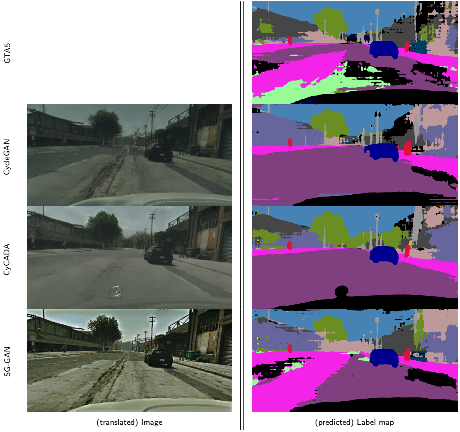
category	GTA5	CycleGAN	CyCADA	SG-GAN
flat	0.740	0.861	**0.894**	0.735
average	0.580	0.560	**0.602**	0.538

class scores

class	GTA5	CycleGAN	CyCADA	SG-GAN
traffic sign	0.090	0.126	0.120	**0.158**
train	0.025	0.115	**0.280**	0.222
average	0.337	0.311	**0.353**	0.326

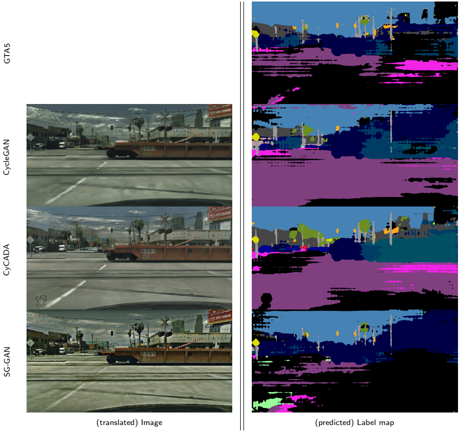
Table: Excerpt of quantitative comparison results (Intersection over Union)

Qualitative



GTA5 (ground-truth)

Qualitative



GTA5 (ground-truth)

Overview

Pre-trained models

- the models used might not represent the ones described in papers
- therefore adaptation not as accurate

DeepLabV3 model

- trained less precise than in its paper
- overall performance scores are lower than expected
- relation between model performance should remain the same though

sampleset of images / techniques with different strengths

- sampleset of images might be too small
- techniques have different strength, e.g. SG-GAN: traffic signs class, CyCADA: train class
- this might alter the average scores

to conclude

- compared three techniques on Synthetic-to-Real Domain Adaptation
- expected all models to increase performance compared to pure GTA5
- CyCADA only one to improve average scores
- SG-GAN improved performance on some classes
- CycleGAN did not perform best for any class or category
- getting the repo code to run was a challenge

Outlook and Future Work

- train models from scratch: better comparison as framework is known exactly
- compare more models: newly developed ones and other existing ones (there might exist better ones)
- other comparison benchmarks: e.g. perceptual loss (compare features in images)
- sampleset size of test images: lower chance of favorable corner cases that increase average score of a specific technique

Thank you

A Comparison of Synthetic-to-Real Domain Adaptation Techniques

Peter Trost

Eberhard Karls Universität Tübingen
Mathematisch-Naturwissenschaftliche Fakultät
Wilhelm-Schickard-Institut für Informatik
Lernbasierte Computer Vision

peter.trost@student.uni-tuebingen.de

October 1, 2019