Feature Generator Layer for Semantic Segmentation Under Different Weather Conditions for Autonomous Vehicles

Özgür Erkent[1], Christian Laugier[1]

Univ. Grenoble Alpes, Inria, Chroma Team, France

Outline

Introduction

Method

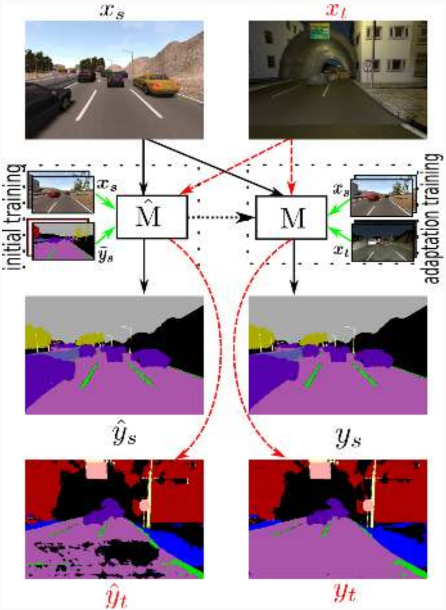
Results

Summary

Unsupervised Domain Adaptation

- No labeled images are used in Unsupervised Domain Adaptation (UDA)
- Suitable for varying weather conditions
- Known Conditions: Source domain, New Conditions: Target Domain
- Our Contributions:
  - Obtain a single model that can classify the surroundings in different weather conditions
  - No labeled image during adaptation training from both domains
  - Adaptation by optimizing only a small set of parameters

Problem Overview



Outline

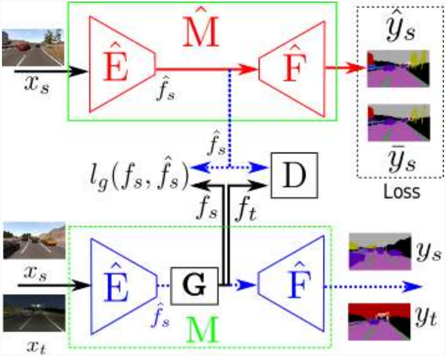
Introduction

Method

Results

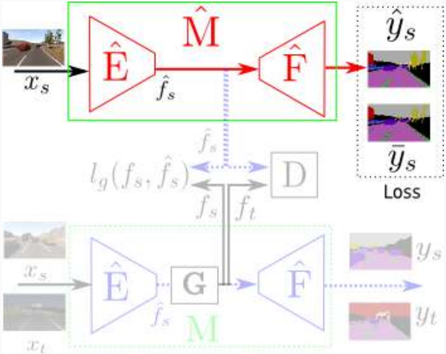
Summary

Method

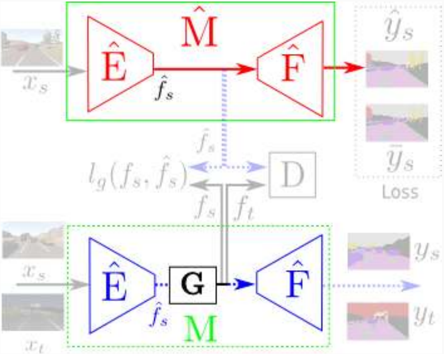


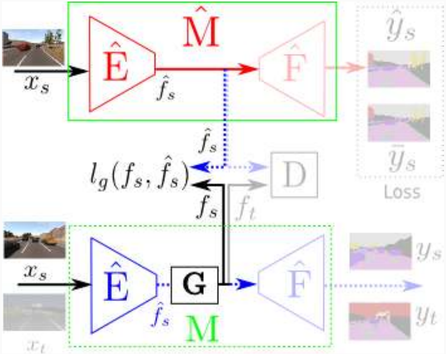
x_s
$\hat{\mathrm{E}}$
$\hat{\mathrm{M}}$
$\hat{f}_s$
$\hat{\mathrm{F}}$
$\hat{y}_s$
$\bar{y}_s$
Loss
$\hat{f}_s$
$l_g(f_s, \hat{f}_s)$
D
f_s
f_t
x_s
x_t
$\hat{\mathrm{E}}$
G
$\hat{f}_s$
M
$\hat{\mathrm{F}}$
y_s
y_t

Method

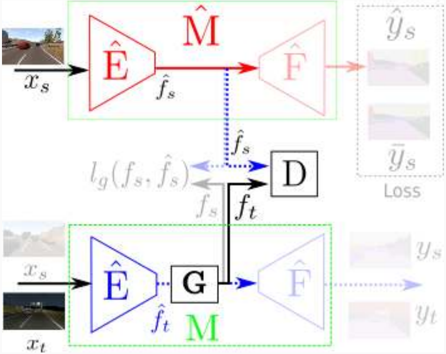


$\hat{M}$
$\hat{E}$
$\hat{F}$
x_s
$\hat{f}_s$
$\hat{y}_s$
$\bar{y}_s$
Loss
$\hat{f}_s$
$l_g(f_s, \hat{f}_s)$
D
f_s
f_t
x_s
x_t
G
M
y_s
y_t


$\hat{M}$
$\hat{E}$
$\hat{F}$
x_s
$\hat{f}_s$
$\hat{y}_s$
$\bar{y}_s$
Loss
$l_g(f_s, \hat{f}_s)$
D
f_s
f_t
x_s
x_t
$\hat{E}$
G
$\hat{F}$
$\hat{f}_s$
M
y_s
y_t

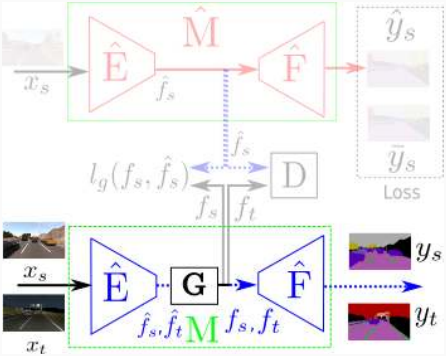

$\hat{M}$
$\hat{E}$
$\hat{F}$
x_s
$\hat{f}_s$
$\hat{y}_s$
$l_g(f_s, \hat{f}_s)$
D
f_s
f_t
Loss
x_s
x_t
G
M
y_s
y_t

Method



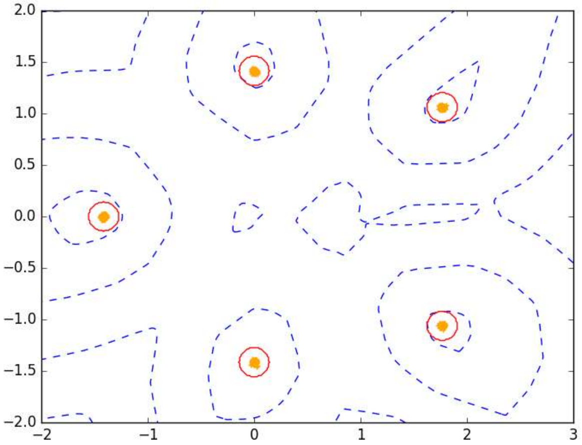
$\hat{\mathrm{M}}$
$\hat{\mathrm{E}}$
$\hat{\mathrm{F}}$
x_s
$\hat{f}_s$
$\hat{y}_s$
$\bar{y}_s$
Loss
$\hat{f}_s$
$l_g(f_s, \hat{f}_s)$
D
f_s
f_t
x_s
x_t
$\hat{\mathrm{E}}$
G
$\hat{\mathrm{F}}$
$\hat{f}_t$
M
y_s
y_t

Method



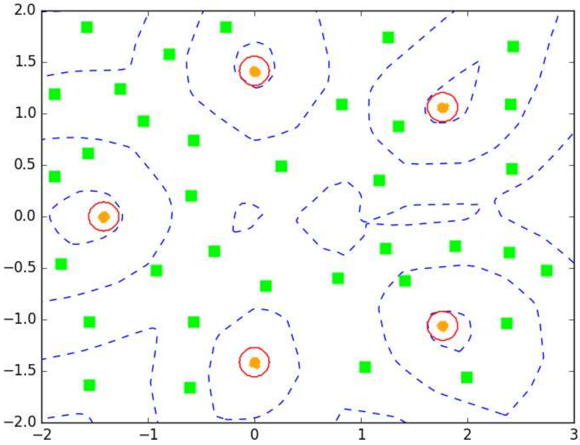
$\hat{M}$
$\hat{E}$
$\hat{F}$
x_s
$\hat{f}_s$
$\hat{y}_s$
$\bar{y}_s$
Loss
$\hat{f}_s$
$l_g(f_s, \hat{f}_s)$
D
f_s
f_t
x_s
x_t
$\hat{E}$
G
$\hat{F}$
$\hat{f}_s, \hat{f}_t$
M
f_s, f_t
y_s
y_t

Intuition for Distance



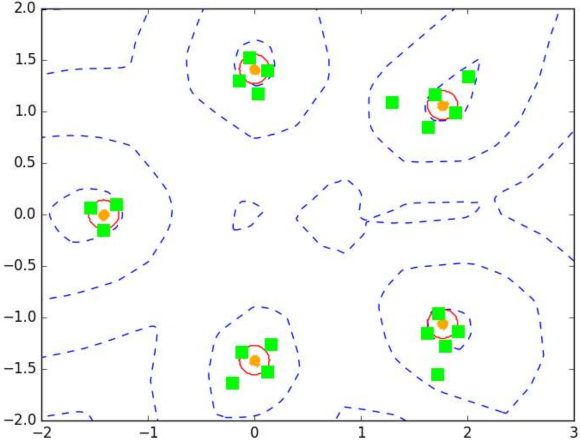
Gaussian Distribution with 5 center in 2D. Now we try to generate data that is similar to this distribution.

Intuition for Distance



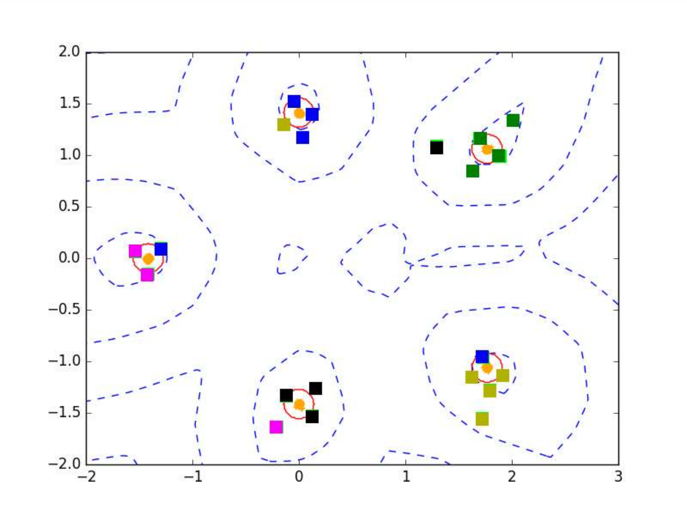
New data don't fit to the distribution, therefore the distribution distance between yellow points and green points D $>>$ 0.

Intuition for Distance



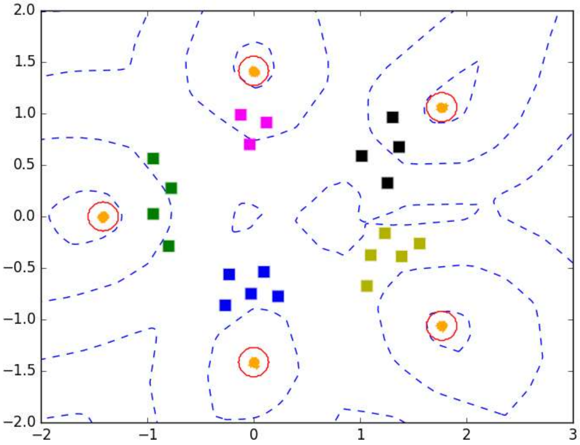
Data fit to the distribution, therefore the distance is small. D≈0.

Intuition for Distance



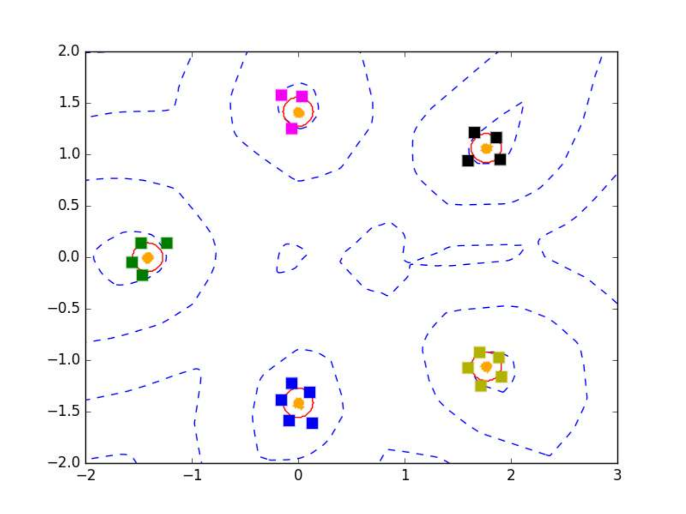
We add color to track the origins of the points.

Intuition for Distance



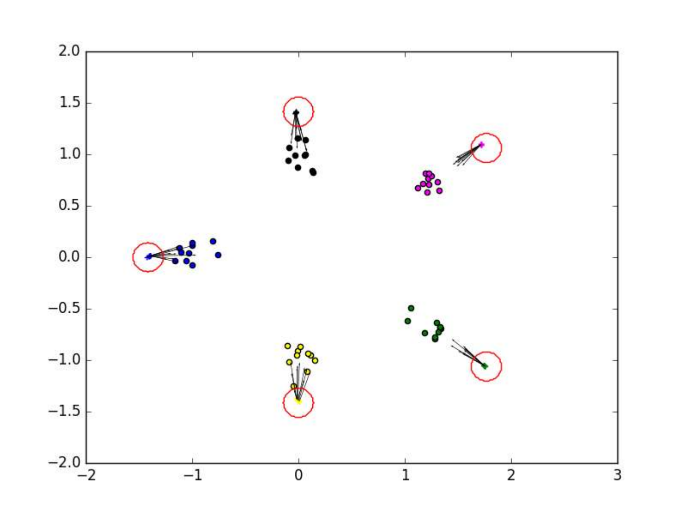
These are the original locations of the points. They don't necessarily converge to the closest Gaussian center.

Intuition for Distance



After adding self-supervision loss, the data converge to the closest center.

Intuition for Distance



Another example. The original locations are shown with circles, the lines represent the direction of motion.

Three alternatives for Similarity Distance

- Self-supervision loss: $\lambda l_g(\hat{\mathbf{f}}_s, \mathrm{G}(\hat{\mathbf{f}}_s))$
- Wasserstein (W-GAN):

$$\inf_{\mathrm{G}(.)\in\mathcal{G}} \inf_{\phi\in\Phi} \mathbb{E}\left[\mathrm{D}_\phi(\hat{\mathbf{f}}_s)\right] - \mathbb{E}\left[\mathrm{D}_\phi(\mathrm{G}(\hat{\mathbf{f}}_t))\right] + \lambda l_g(\hat{\mathbf{f}}_s, \mathrm{G}(\hat{\mathbf{f}}_s))$$

- Jensen-Shannon Distance (GAN):

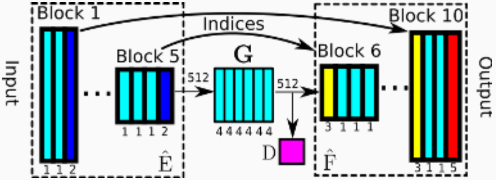
$$\inf_{\mathrm{G}(.)\in\mathcal{G}} \inf_{\phi\in\Phi} \mathbb{E}\left[\log(\mathrm{D}_\phi(\hat{\mathbf{f}}_s))\right] + \mathbb{E}\left[\log(1 - \mathrm{D}_\phi(\mathrm{G}(\hat{\mathbf{f}}_s)))\right] + \lambda l_g(\hat{\mathbf{f}}_s, \mathrm{G}(\hat{\mathbf{f}}_s))$$

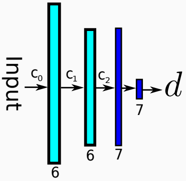
- Maximum-Mean Discrepancy (MMD):

$$\begin{aligned} \mathrm{MMD}_k(\mathrm{Pr}_p, \mathrm{Pr}_q) = \mathbb{E}_{\hat{\mathbf{f}}_s, \hat{\mathbf{f}}_{s'}} k(\hat{\mathbf{f}}_s, \hat{\mathbf{f}}_{s'}) + \mathbb{E}_{\mathbf{f}_t, \mathbf{f}_{t'}} k(\mathbf{f}_t, \mathbf{f}_{t'}) \\ - 2\mathbb{E}_{\hat{\mathbf{f}}_s, \mathbf{f}_t} k(\hat{\mathbf{f}}_s, \mathbf{f}_t) \end{aligned} \tag{1}$$

Implementation in SegNet

Segnet, Discriminator.





Outline

Introduction

Method

Results

Summary

Quantitative Evaluation

- Tested on SYNTHIA Dataset: Synthetic dataset with images from a simulator.
- Various weather and driving conditions.
- Selected conditions: Spring (Source Domain), Winter, Night.
- For Winter-01, the images are observed and adapted. (same for Night)
- For Winter-06, the adapted model to Winter-01 is used; no additional adaptation training. (same for Night)

Quantitative Results

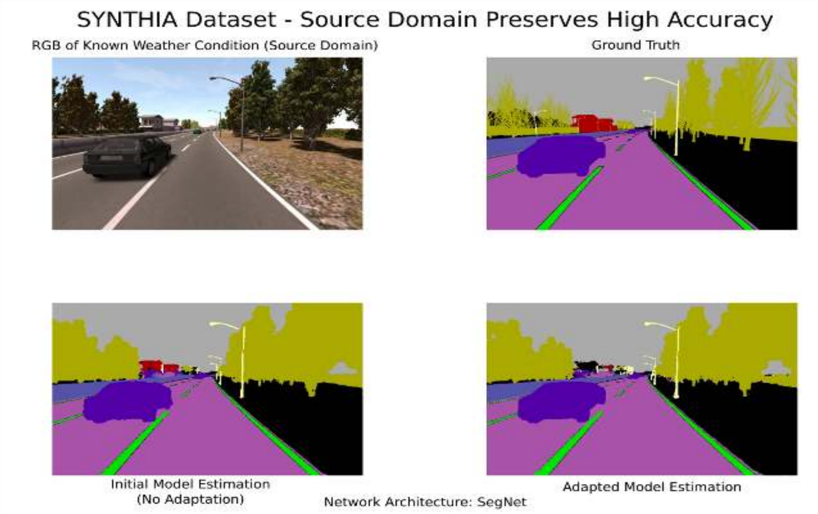
Table 1: Adaptation to Winter for SYNTHIA with SegNet

% mIoU	Init	FGL+W1	FGL+JS	FGL+MMD
Spring-06	**59.8(0)**	59.4(-0.7)	59.5(-0.4)	59.7(-0.2)
Winter-01	55.9(0)	**61.7(10.4)**	60.6(8.4)	59.6(6.6)
Winter-06	47.8(0)	48.5(1.4)	48.5(1.4)	48.5(1.5)

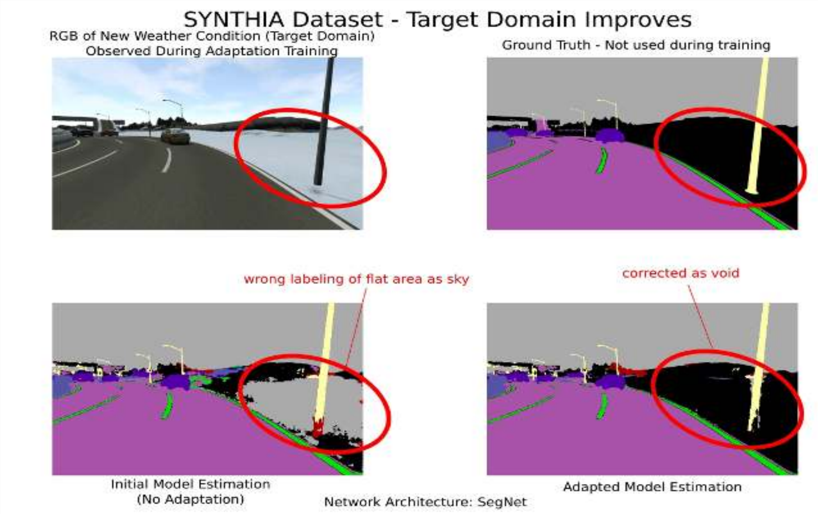
Table 2: Adaptation to Night for SYNTHIA with SegNet

% mIoU	Init	FGL+W1	FGL+JS	FGL+MMD
Spring-06	59.8(0)	59.7(-0.1)	59.9(0.2)	**60.0(0.4)**
Night-01	53.7(0)	**58.6(9.1)**	58.2(8.4)	57.5(7.0)
Night-06	35.6(0)	**38.8(8.9)**	38.3(7.6)	38.4(7.8)

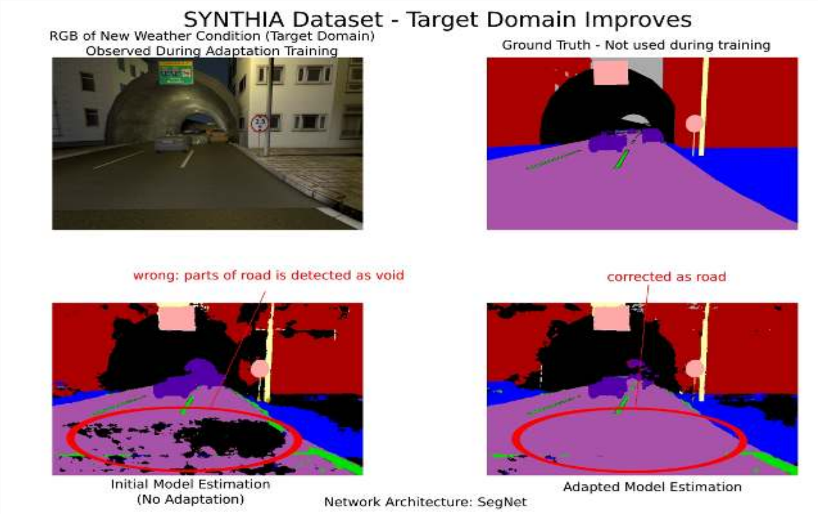
Qualitative Evaluation - Synthia Dataset



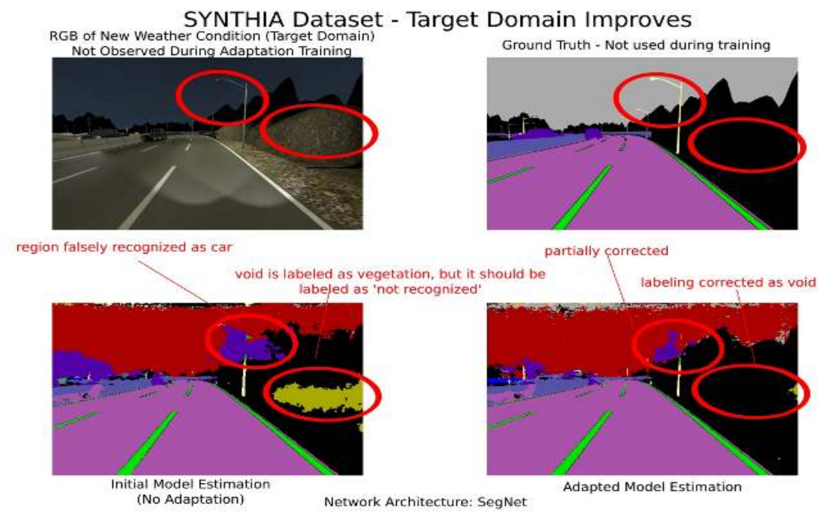
Qualitative Evaluation - Synthia Dataset



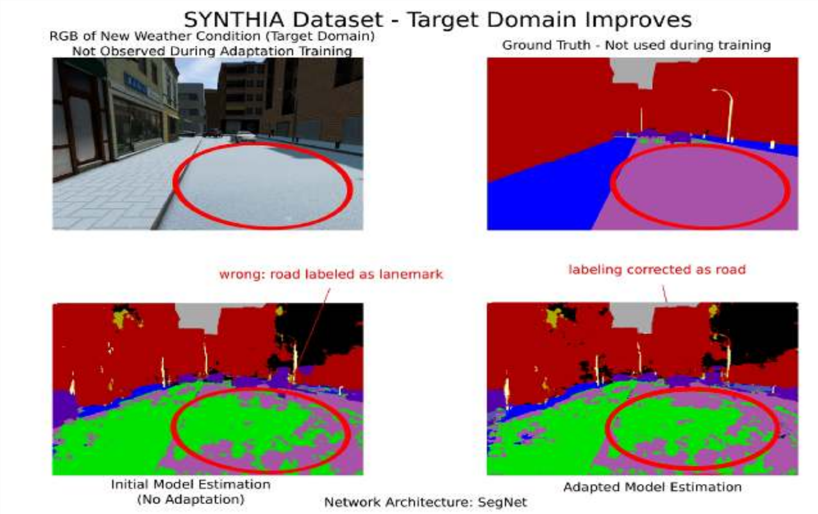
Qualitative Evaluation - Synthia Dataset



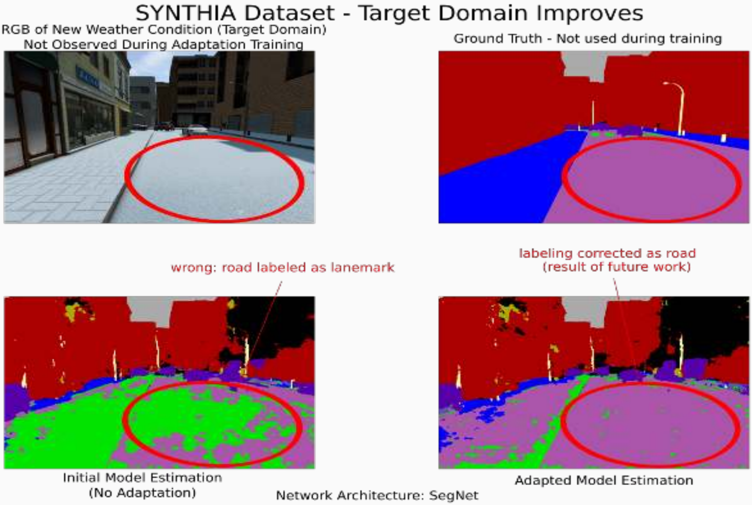
Qualitative Evaluation - Synthia Dataset



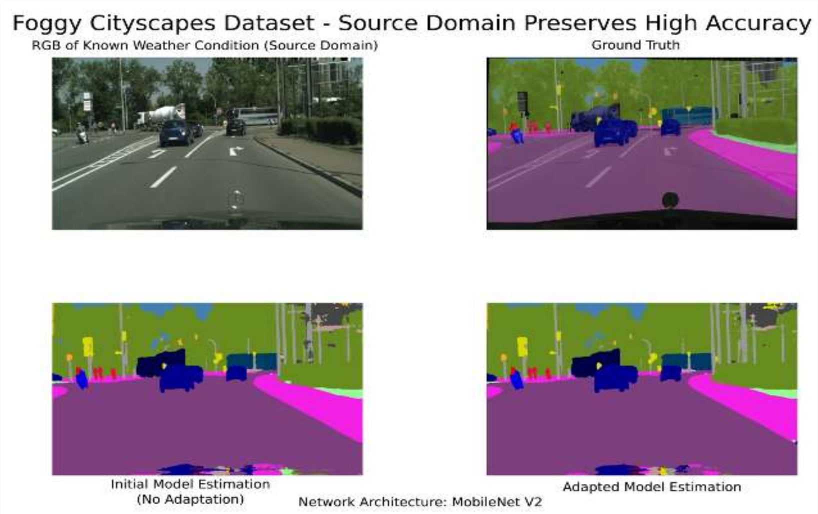
Qualitative Evaluation - Synthia Dataset



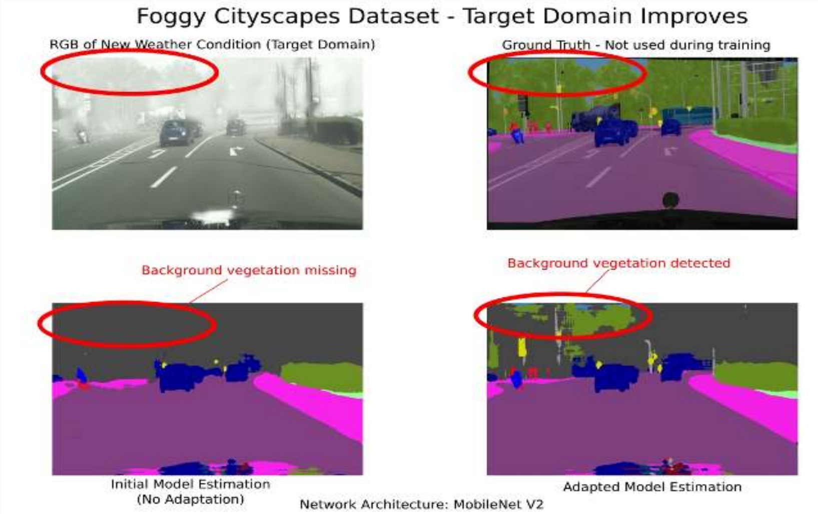
Qualitative Evaluation - Synthia Dataset



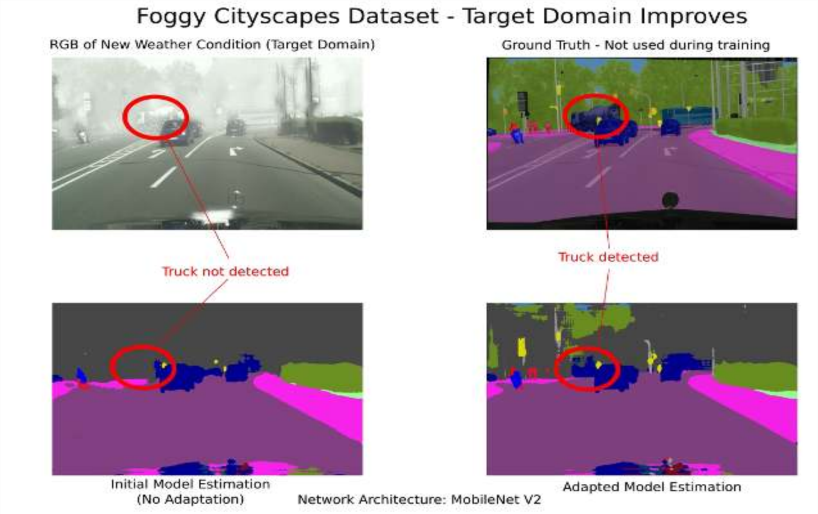
Qualitative Evaluation - Future Work - Cityscapes Dataset



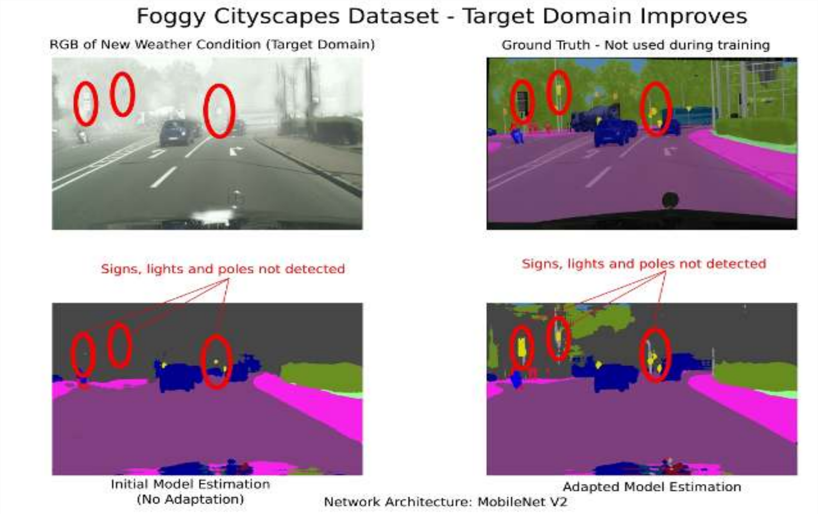
Qualitative Evaluation - Future Work - Cityscapes Dataset



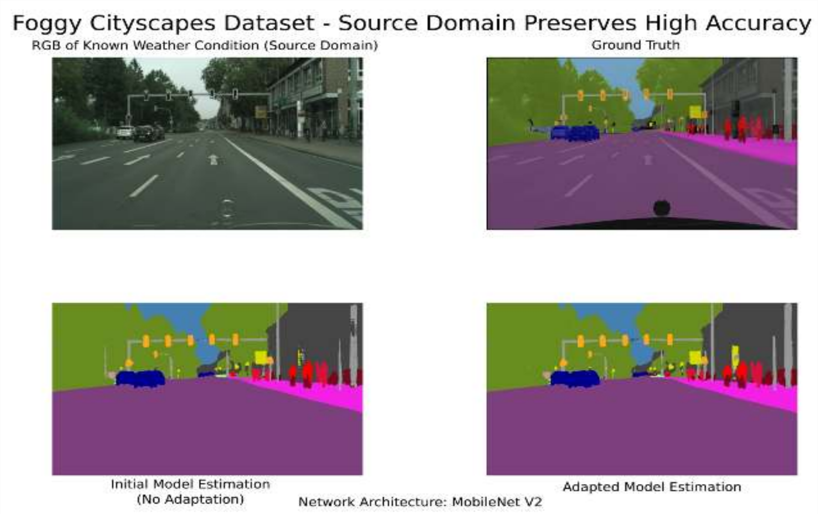
Qualitative Evaluation - Future Work - Cityscapes Dataset



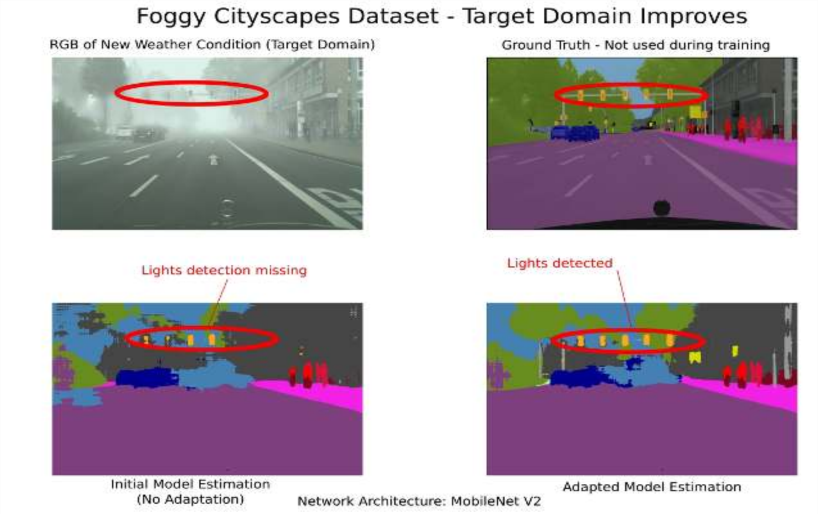
Qualitative Evaluation - Future Work - Cityscapes Dataset



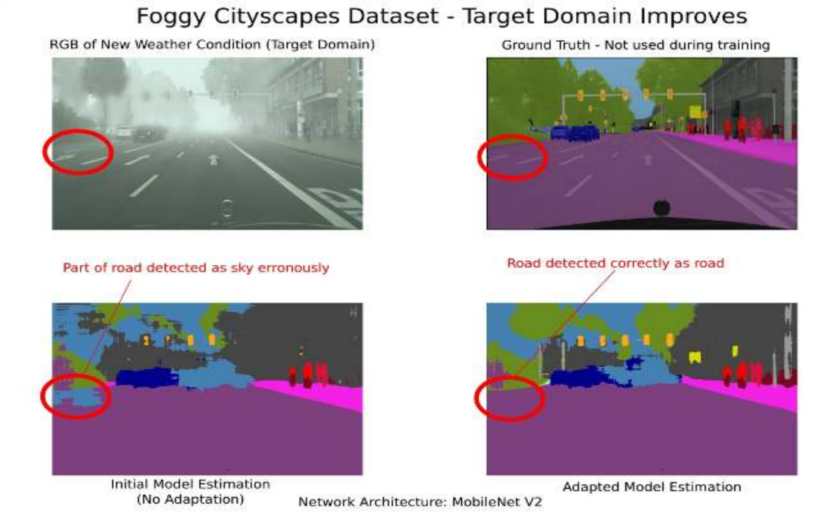
Qualitative Evaluation - Future Work - Cityscapes Dataset



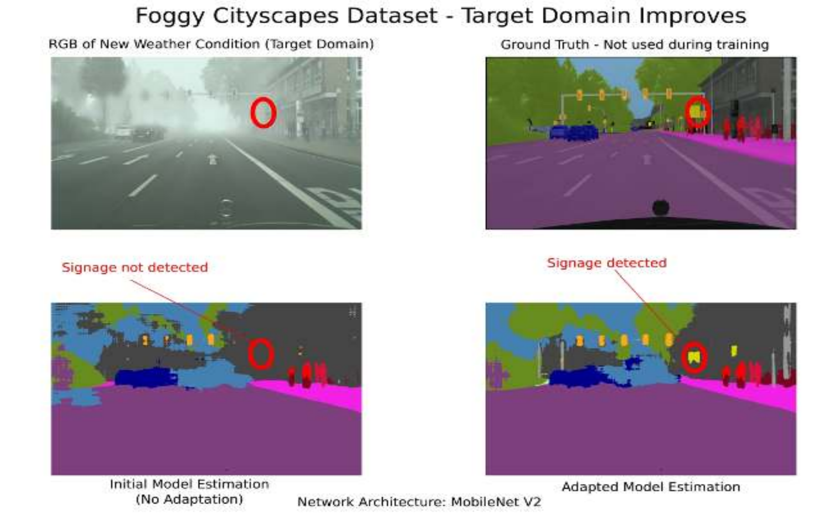
Qualitative Evaluation - Future Work - Cityscapes Dataset



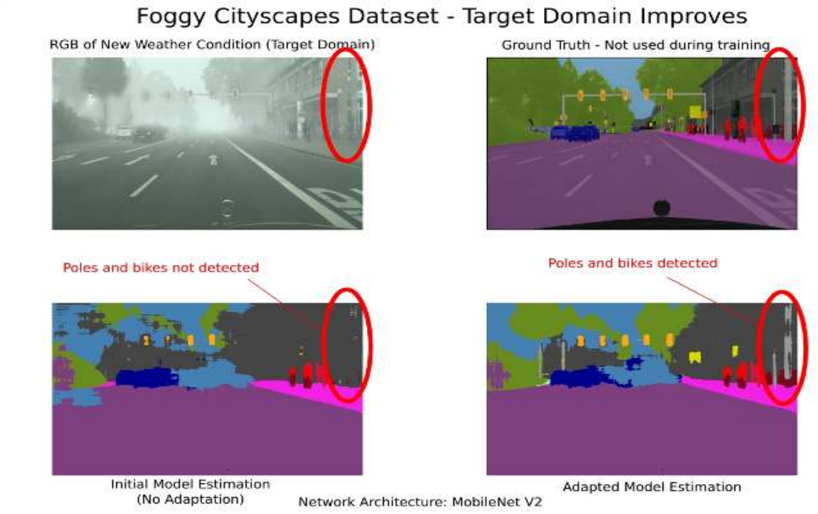
Qualitative Evaluation - Future Work - Cityscapes Dataset



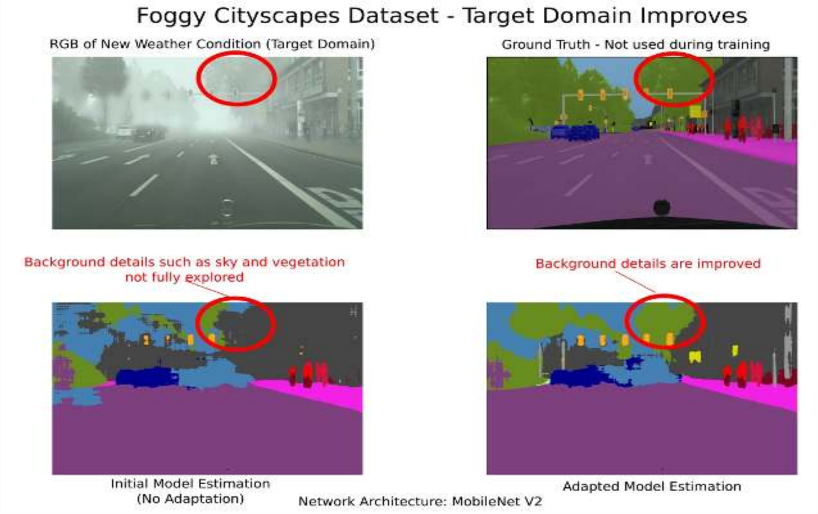
Qualitative Evaluation - Future Work - Cityscapes Dataset



Qualitative Evaluation - Future Work - Cityscapes Dataset



Qualitative Evaluation - Future Work - Cityscapes Dataset



KITTI Dataset

- 200 Training, 200 Test images.
- 19 class labels, similar annotation with Cityscapes.

Late Breaking Result from KITTI Dataset

Better than 10 supervised methods,
Better than the other adapted model.

	Method	Setting	Code	IoU class	iIoU class	IoU category	iIoU category	Runtime	Environment	Compare
1	VideoProp-LabelRelax			72.82	48.68	88.99	75.26	n s	GPU @ 1.5 Ghz (Python + C/C++)	
	Y. Zhu*: Improving Semantic Segmentation via Video Propagation and Label Relaxation. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2019.									
2	DH-MDS			71.40	39.66	87.00	63.07	1.3 s	1 core @ 2.5 Ghz (Python)	
3	[illegible]			[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	
4	Mapillary_ROB			[illegible]	41.64	86.06	[illegible]	[illegible]	1 core @ 2.5 Ghz (Python + C/C++)	
5	LDN2_ROB		code	63.51	28.31	85.34	59.07	1 s	GPU @ 2.5 Ghz (C/C++)	
6	AHiSS_ROB			61.24	26.94	81.54	53.42	0.06 s	GPU @ 1.5 Ghz (Python)	
7	Chroma UDA			60.36	31.70	80.73	61.91	0.4 s	GPU @ 2.5 Ghz (Python)	
8	IfN-DomAdap-Seg			59.50	30.28	81.57	61.91	1 s	GPU @ 2.0 Ghz (Python)	
	J. Bolte, M. Kamp, A. Breuer, S. Homoceanu, P. Schlicht, F. Hüger, D. Lipinski and T. Fingscheidt: Unsupervised Domain Adaptation to Improve Image Segmentation Quality Both in the Source and Target Domain. [illegible]									
9	SegStereo		code	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	Nvidia GTX Titan Xp	
	G. Yang, H. [illegible], J. Shi, Z. Deng and J. Jia: SegStereo: Exploiting Semantic Information for Disparity Estimation. ECCV 2018.									
10	[illegible]			[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	
11	SJTU_HHW			[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	GPU @ 2.5 Ghz (Python)	
12	AdapNetv2_ROB			54.97	25.20	81.64	56.31	1 s	1 core @ 2.5 Ghz (C/C++)	
13	VlocNet++_ROB			53.92	23.68	80.74	53.66	n s	1 core @ 2.5 Ghz (C/C++)	
14	HiSS_ROB			53.16	21.37	78.32	51.92	0.06 s	GPU @ 1.5 Ghz (Python)	
15	SDNet		code	51.14	17.74	79.62	50.45	0.2 s	GPU @ 2.5 Ghz (Python + C/C++)	
	M. Ochs, A. Kretz and R. Mester: SDNet: Semantic Guided Depth Estimation Network. German Conference on Pattern Recognition (GCPR) 2019.									
16	APMoE_seg_ROB		code	47.96	17.86	78.11	49.17	0.2 s	GPU @ 3.5 Ghz (Matlab/C++)	
	S. Kong and C. Fowlkes: Pixel-wise Attentional Gating for Parsimonious Pixel Labeling. arxiv 1805.01556 2018.									
17	BatMAN_ROB			47.36	16.79	78.43	50.03	0.2 s	GPU @ 1.5 Ghz (Python)	
18	FCN101_ROB			24.57	6.19	51.85	22.81	0.07 s	2 cores @ 3.5 Ghz (Python)	

Without adaptation, the model would be on the 17th place. There exists one more model with adaptation at 8th place.

- Adaptation from cityscapes labels.
- No labels from KITTI dataset are used for learning.
- Further details in the future work.
- We are better than 10 other supervised methods, and better than the other only adaptation method.

Outline

Introduction

Method

Results

Summary

Summary

- A method that can be used in different weather conditions during the same ride.
- No source labels during adaptation training thanks to self-supervised loss.
- Fast training since only a portion of the parameters are updated.