# Deep Scene Understanding from Images

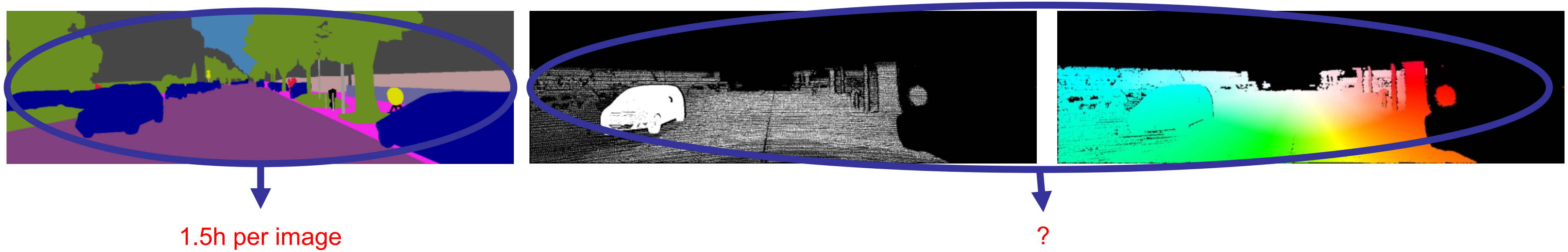
Matteo Poggi, Fabio Tosi, **Pierluigi Zama Ramirez**
Computer Vision Lab (CVLab), University of Bologna

# 3 – Unsupervised Domain Adaptation for Semantic Segmentation

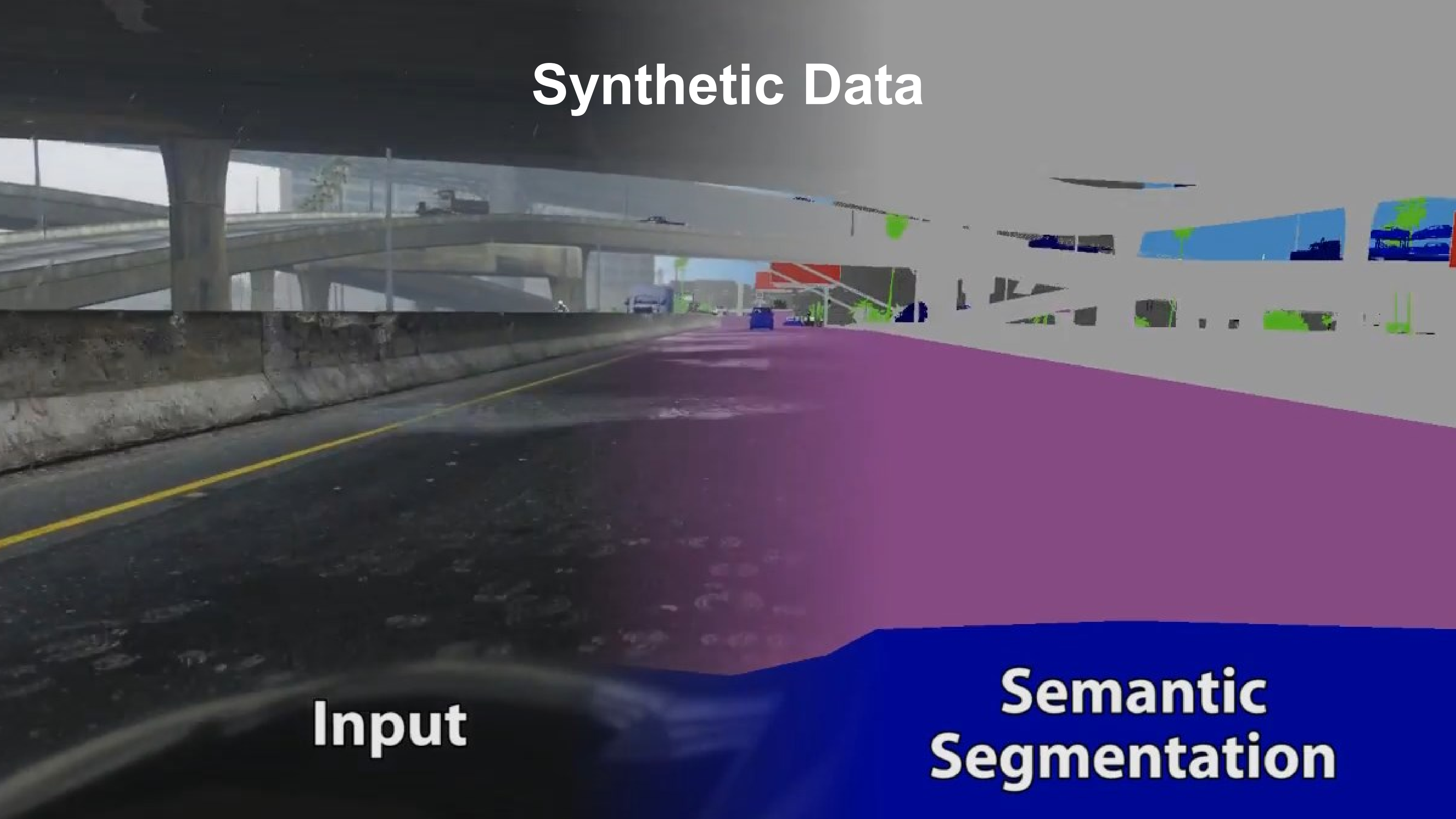
# Data Problem

## What if we lack labels?

Annotated data are extremely difficult to obtain. For instance, for semantic segmentation, we need several hours to manually annotate a single image. For depth and optical flow, manually labelling is almost impossible.



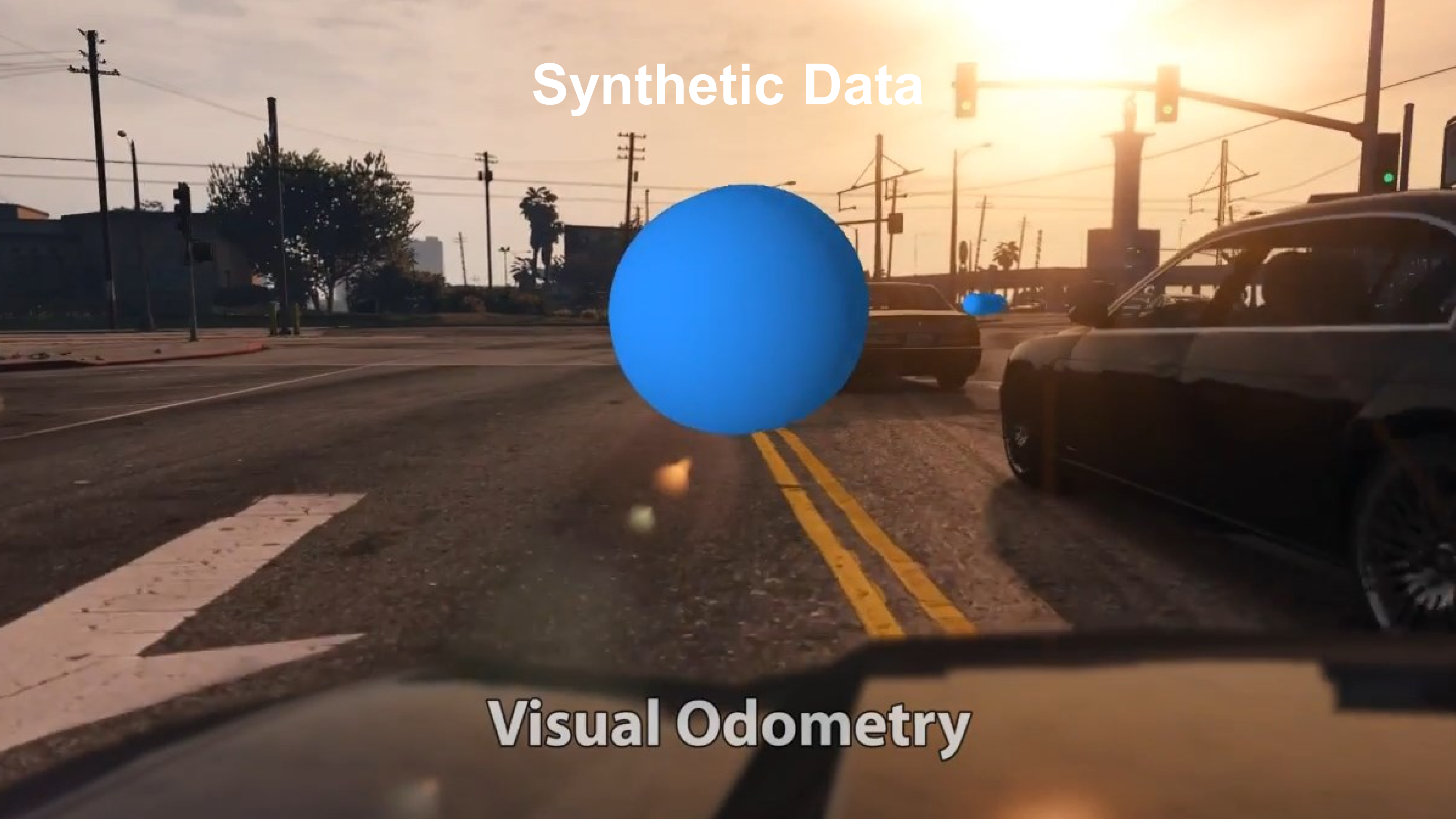
# Synthetic Data




Synthetic Da
Input
Semantic Instance
Segmentation

# Synthetic Da

# Synthetic Data



# Domain Shift

Synthetic vs Real Data

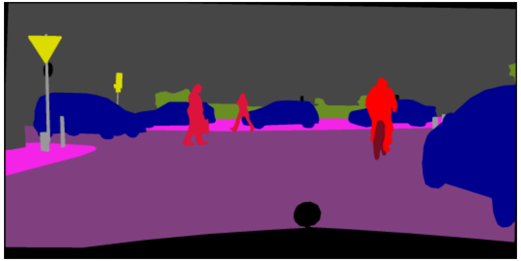
Do you note any differences?

Colors | Textures | Light | Sensor Noise | Object Shapes | Class Frequency | Object/Camera Positioning | …

# Domain Shift

Source Training Distribution ≠ Target Test Distribution

Real Image

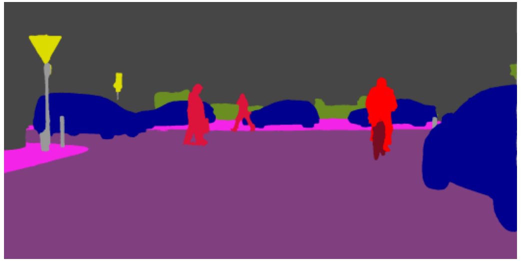


Manually annotated image

Network trained on synthetic data only

Performance gap

**Domain Adaptation**



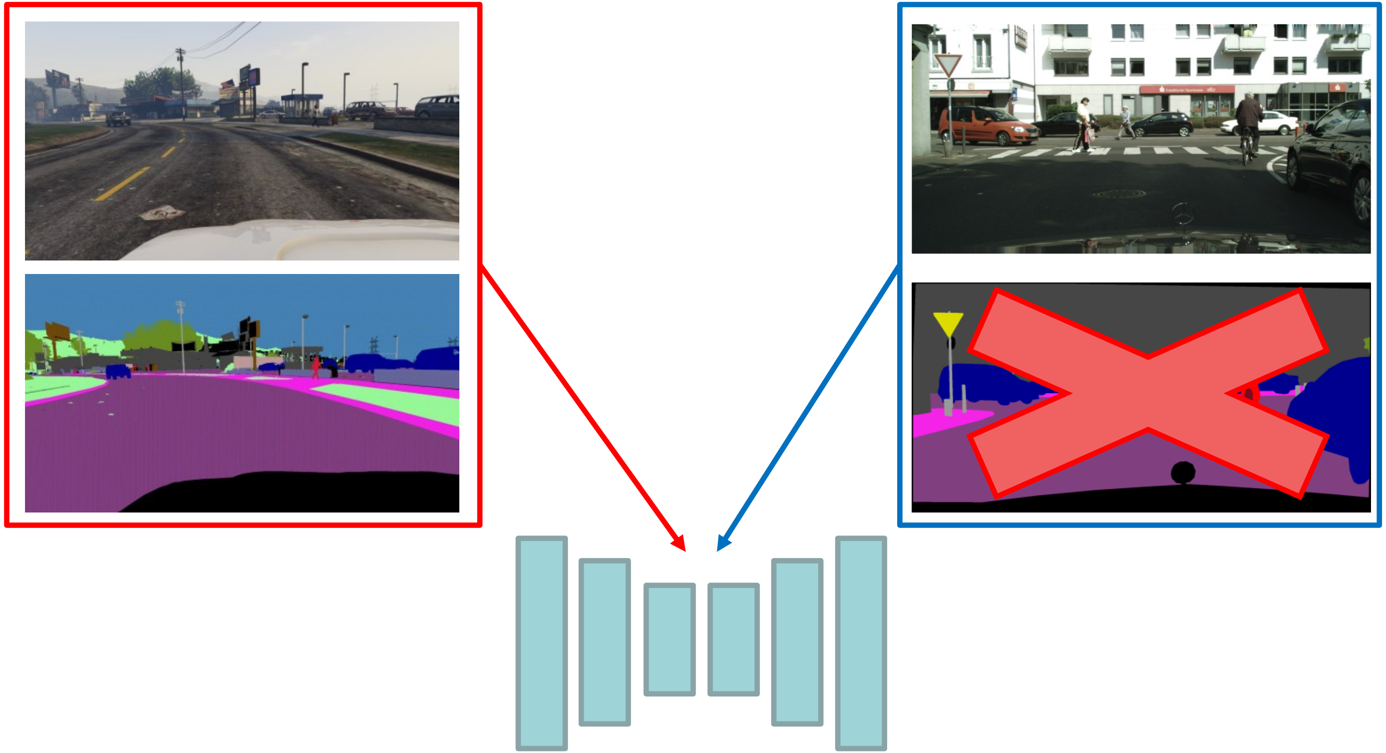
Network trained on real data

# Unsupervised Domain Adaptation

Source Domain (Synthetic)

Target Domain (Real)

How can we exploit them?



# Some Benchmarks for UDA for Semantic Segmentation

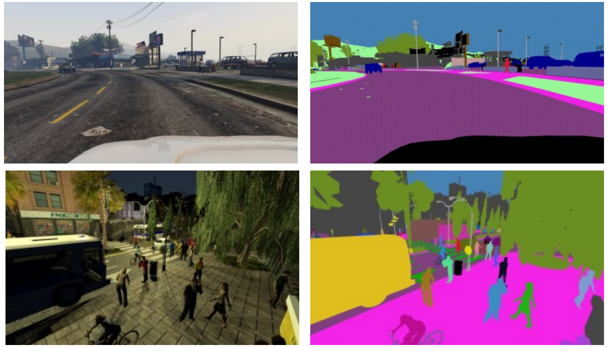
**Most Popular!**

**Urban Environment, Autonomous Driving**

**Synthetic → Real**

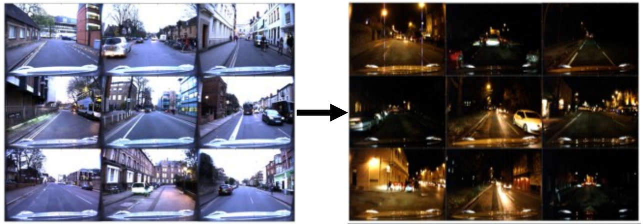
GTAV

Synthia



Cityscapes

Day → Night



Cityscapes → Across Cities

Switzerland/Germany

Rome

Rio

Tokyo

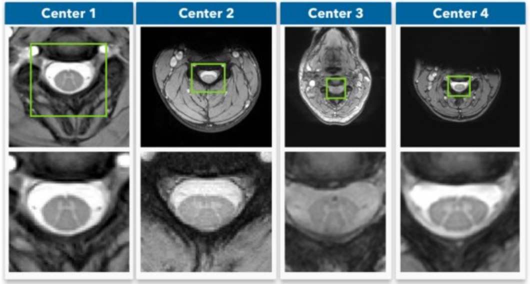
Taipe

**Aerial Images Across Cities**

| Inria Dataset | Massachusetts Dataset | WHU Dataset | Lee et al. Dataset |
|---|---|---|---|

**MRI of different centers**

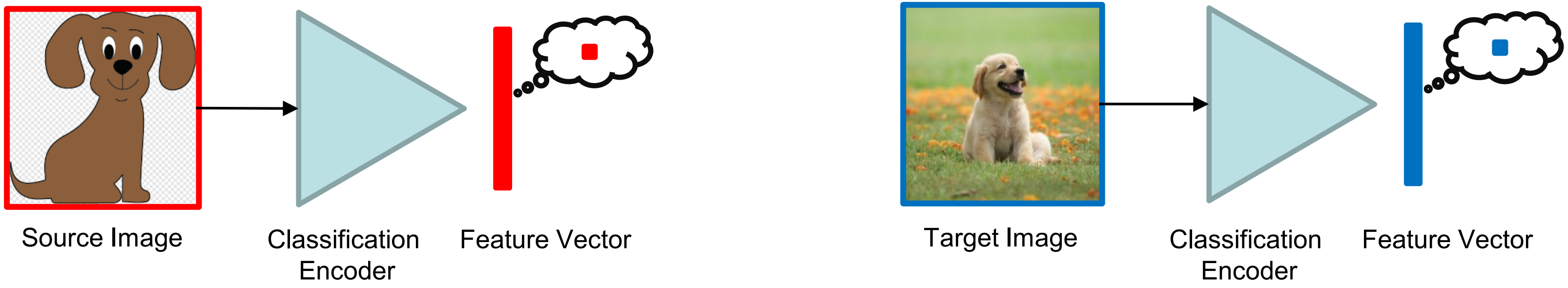
UCL | Montreal | Zurich | Vanderbilt




Gray Matter (GM) segmentation challenge (Prados et al., 2017).

# Domain Alignment

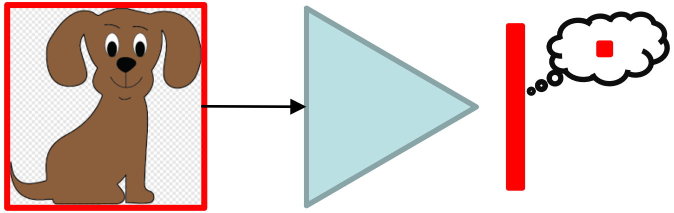
# Domain Alignment




Red: Source samples
Blue: Target samples

Chen, C., Chen, Z., Jiang, B., & Jin, X. (2019). Joint Domain Alignment and Discriminative Feature Learning for Unsupervised Deep Domain Adaptation. *AAAI*.

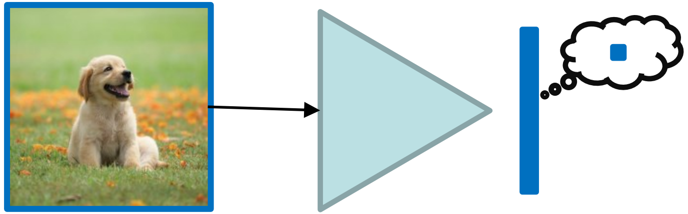
# Domain Alignment



Source Image

Classification Encoder

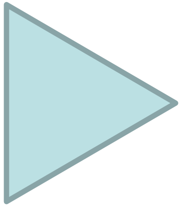
Feature Vector



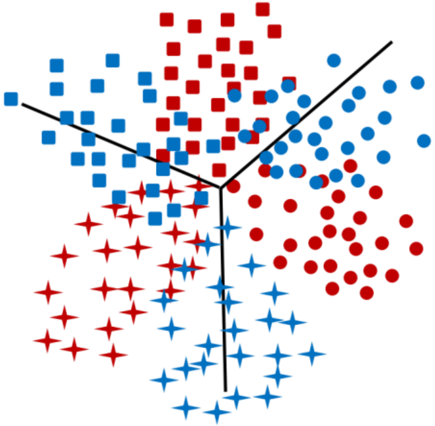
Target Image

Classification Encoder

Feature Vector



**Trained Only on Source Domain**



**Feature Space**

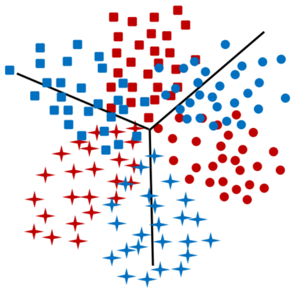
Red: Source samples
Blue: Target samples
**Black line**: Hyperplane learned from the source domain

*Circle, Square* and *Star* indicate three different categories, respectively

Chen, C., Chen, Z., Jiang, B., & Jin, X. (2019). Joint Domain Alignment and Discriminative Feature Learning for Unsupervised Deep Domain Adaptation. *AAAI*.

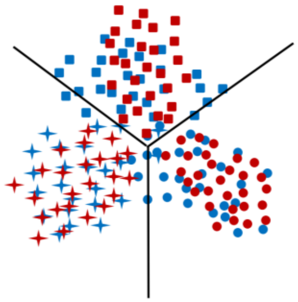
# Domain Alignment

**Trained Only on Source Domain**



**Encoder Feature Space**

**Trained with Domain Alignment**



Red: Source samples
Blue: Target samples
**Black line**: Hyperplane learned from the source domain

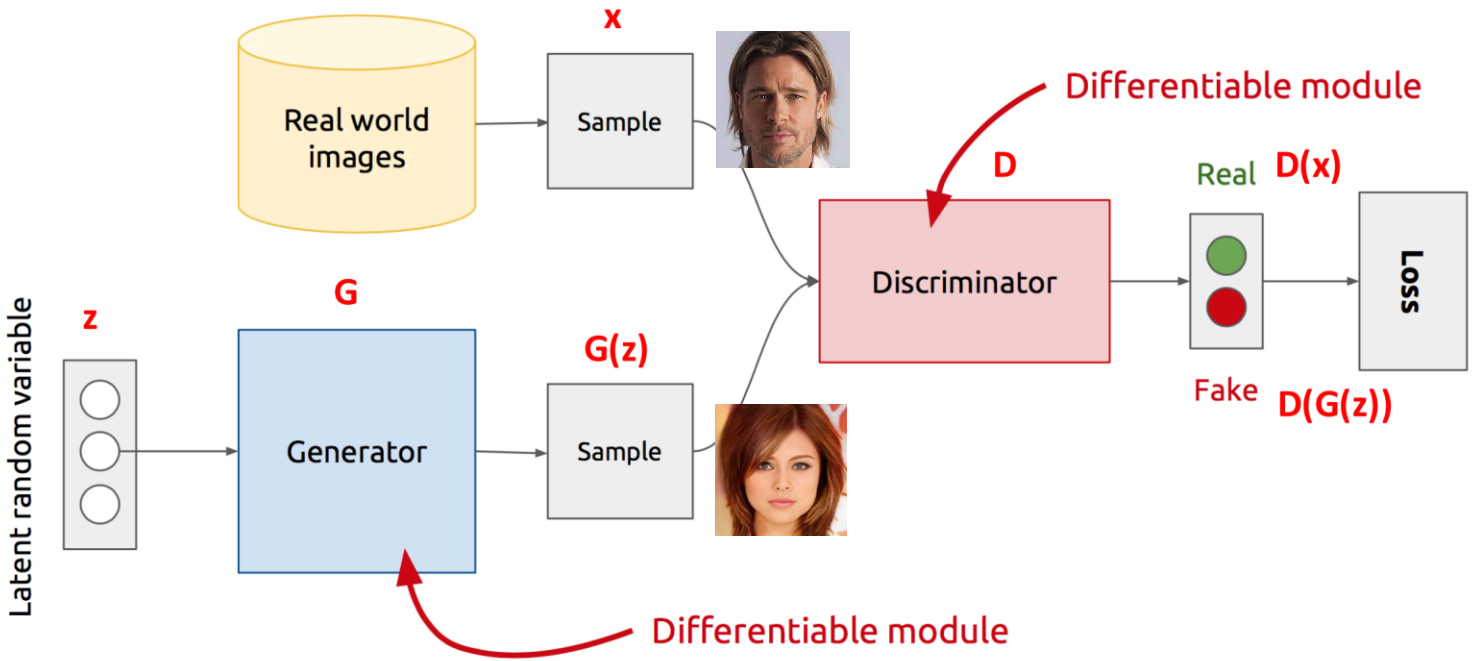
*Circle*, *Square* and *Star* indicate three different categories, respectively

**How can we achieve it?**

- **Adversarial**
- Maximum Mean Discrepancy (MMD)
- Optimal Transport
- ...

Chen, C., Chen, Z., Jiang, B., & Jin, X. (2019). Joint Domain Alignment and Discriminative Feature Learning for Unsupervised Deep Domain Adaptation. *AAAI*.

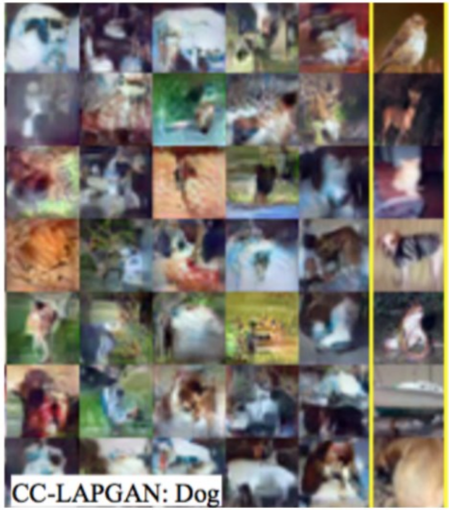
# Generative Adversarial Networks (GANs)




Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, *27*.

# Generative Adversarial Networks (GANs)

2009

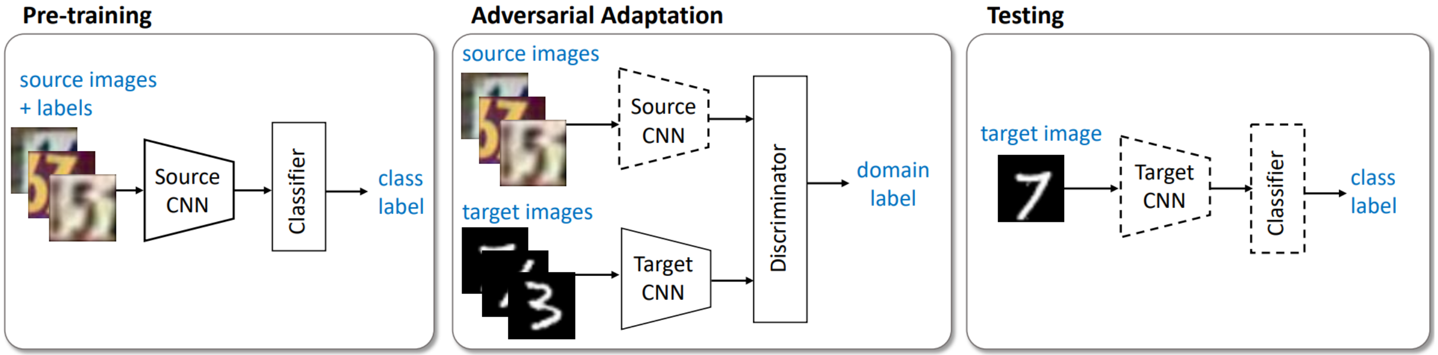



2015

2018

Karras, T., Laine, S., & Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4401-4410).

# Adversarial Discriminative Domain Adaptation (ADDA)




1 - Pre-train a source encoder CNN using labeled source image examples.

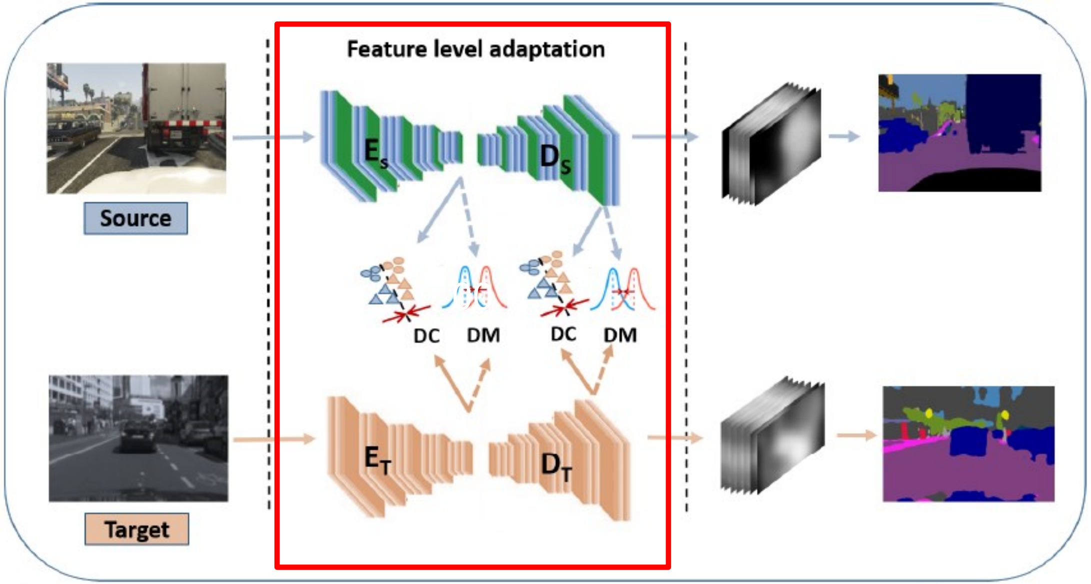
2 - Perform adversarial adaptation by learning a target encoder CNN such that a discriminator that sees encoded source and target examples cannot reliably predict their domain label.

3 - During testing, target images are mapped with the target encoder to the shared feature space and classified by the source classifier. Dashed lines indicate fixed network parameters

Tzeng, E., Hoffman, J., Saenko, K., & Darrell, T. (2017). Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7167-7176).

# Domain Alignment in Semantic Segmentation

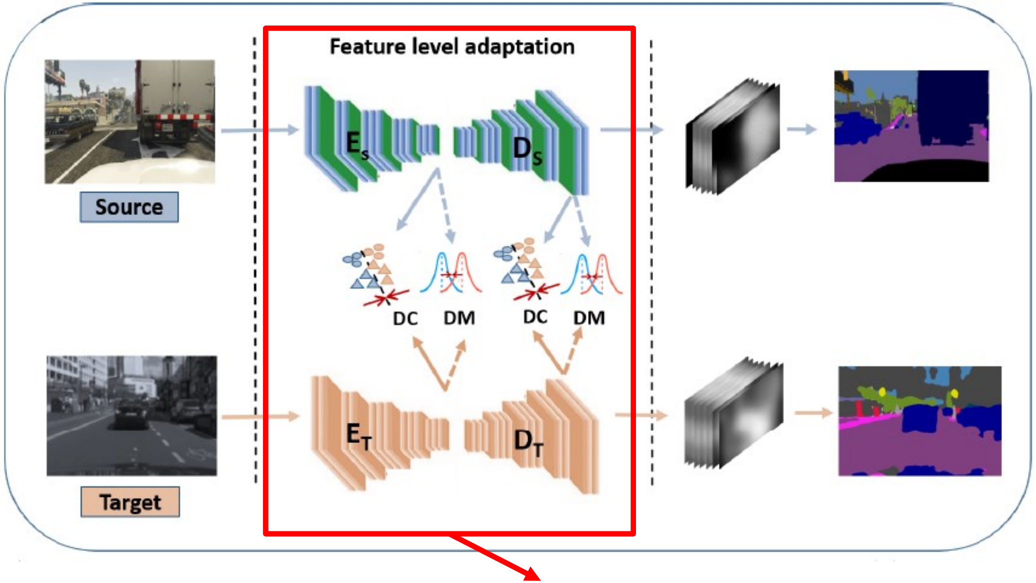
## Feature Level



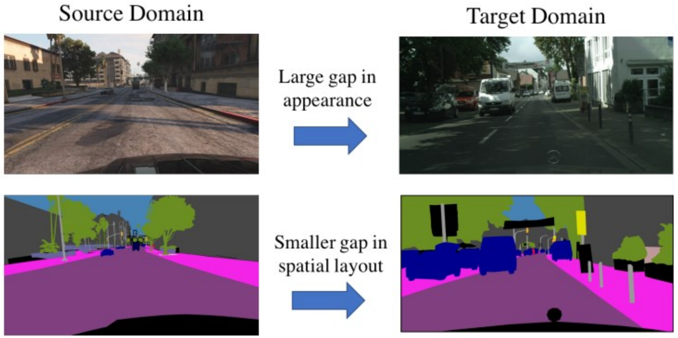

In generic DA, domain alignment is often performed in a single latent representation space.
In Semantic Segmentation networks, the alignment is often done **at multiple layers**, by discrepancy minimization between feature distributions or by adversarial learning relying on a domain classifier (DC) to increase domain confusion.
Encoders and decoders of the segmentation network are often shared.

# Domain Alignment

## Output Level

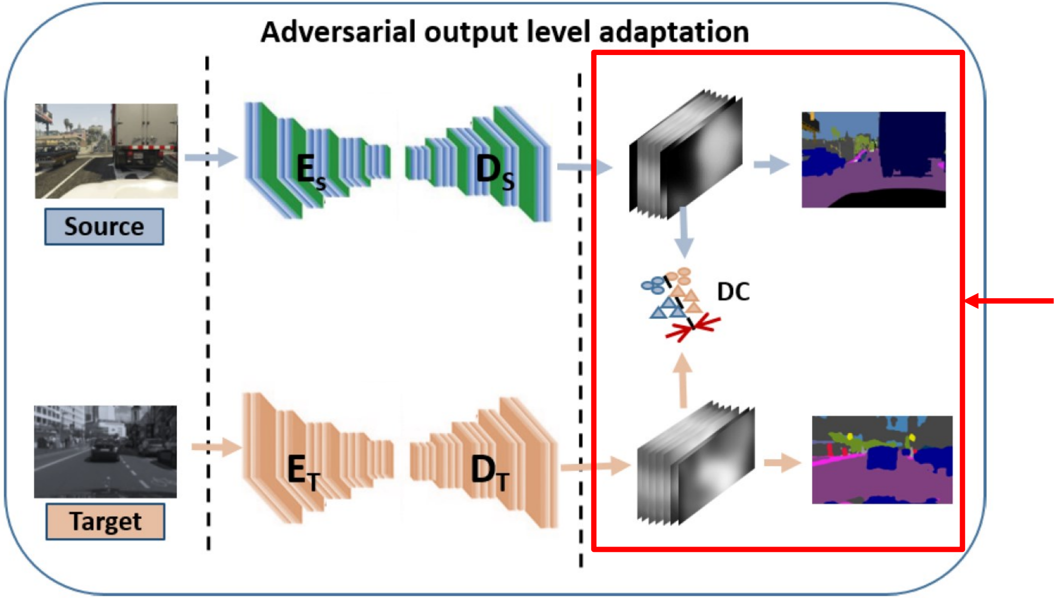


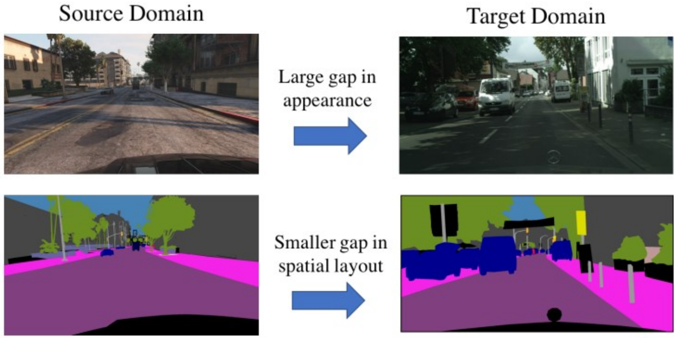

However, differently from the image classification task, feature adaptation for semantic segmentation may suffer from the complexity of high-dimensional features that needs to encode diverse visual cues,  including appearance, shape and context.




While images may be very different in appearance, their outputs are structured and share many similarities, such as spatial layout and local context.

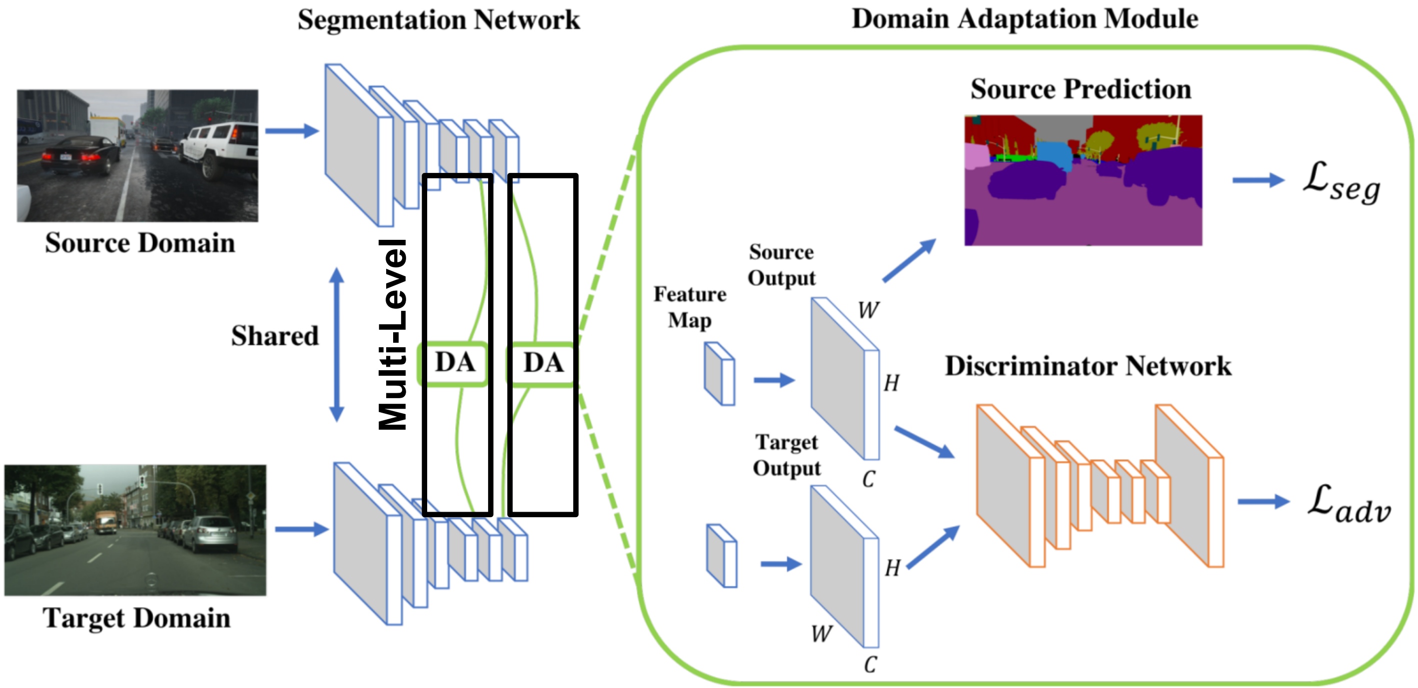
# Domain Alignment
## Output Level




While images may be very different in appearance, their outputs are structured and share many similarities, such as spatial layout and local context.

# Domain Alignment
## Output Level




Tsai, Y. H., Hung, W. C., Schulter, S., Sohn, K., Yang, M. H., & Chandraker, M. (2018). Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7472-7481).

# Domain Alignment
## Output Level

GTA5 to Cityscapes

### Feature-Level

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Baseline | | | | | | | | | | | | | | | | | | | | 17.9 |
| FCNs in the Wild [13] | 70.4 | 32.4 | 62.1 | 14.9 | 5.4 | 10.9 | 14.2 | 2.7 | 79.2 | 21.3 | 64.6 | 44.1 | 4.2 | 70.4 | 8.0 | 7.3 | 0.0 | 3.5 | 0.0 | 27.1 |
| CyCADA (feature) [12] | 85.6 | 30.7 | 74.7 | 14.4 | 13.0 | 17.6 | 13.7 | 5.8 | 74.6 | 15.8 | 69.9 | 38.2 | 3.5 | 72.3 | 16.0 | 5.0 | 0.1 | 3.6 | 0.0 | 29.2 |

### Output-Level

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Ours (singel-level) | **87.3** | 29.8 | **78.6** | **21.1** | **18.2** | **22.5** | 21.5 | 11.0 | 79.7 | **29.6** | **71.3** | 46.8 | 6.5 | **80.1** | **23.0** | **26.9** | 0.0 | 10.6 | 0.3 | **35.0** |

### Ablation study

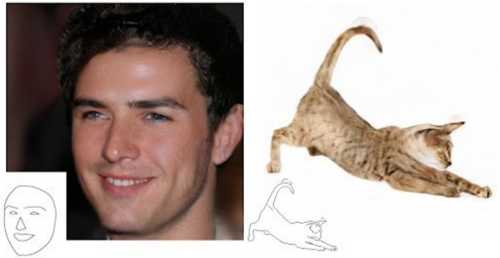
| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Baseline (ResNet) | 75.8 | 16.8 | 77.2 | 12.5 | 21.0 | 25.5 | 30.1 | 20.1 | 81.3 | 24.6 | 70.3 | 53.8 | 26.4 | 49.9 | 17.2 | 25.9 | 6.5 | 25.3 | **36.0** | 36.6 |
| Ours (feature) | 83.7 | 27.6 | 75.5 | 20.3 | 19.9 | **27.4** | 28.3 | **27.4** | 79.0 | 28.4 | 70.1 | 55.1 | 20.2 | 72.9 | 22.5 | **35.7** | **8.3** | 20.6 | 23.0 | 39.3 |
| Ours (single-level) | **86.5** | 25.9 | 79.8 | 22.1 | 20.0 | 23.6 | 33.1 | 21.8 | 81.8 | 25.9 | **75.9** | 57.3 | 26.2 | **76.3** | 29.8 | 32.1 | 7.2 | 29.5 | 32.5 | 41.4 |
| Ours (multi-level) | **86.5** | **36.0** | **79.9** | **23.4** | **23.3** | 23.9 | **35.2** | 14.8 | **83.4** | **33.3** | 75.6 | **58.5** | **27.6** | 73.7 | **32.5** | 35.4 | 3.9 | **30.1** | 28.1 | **42.4** |
| Oracle | | | | | | | | | | | | | | | | | | | | 65.1 |

# Image to Image Translation


Style Transfer


Image Synthesis

Image Inpainting



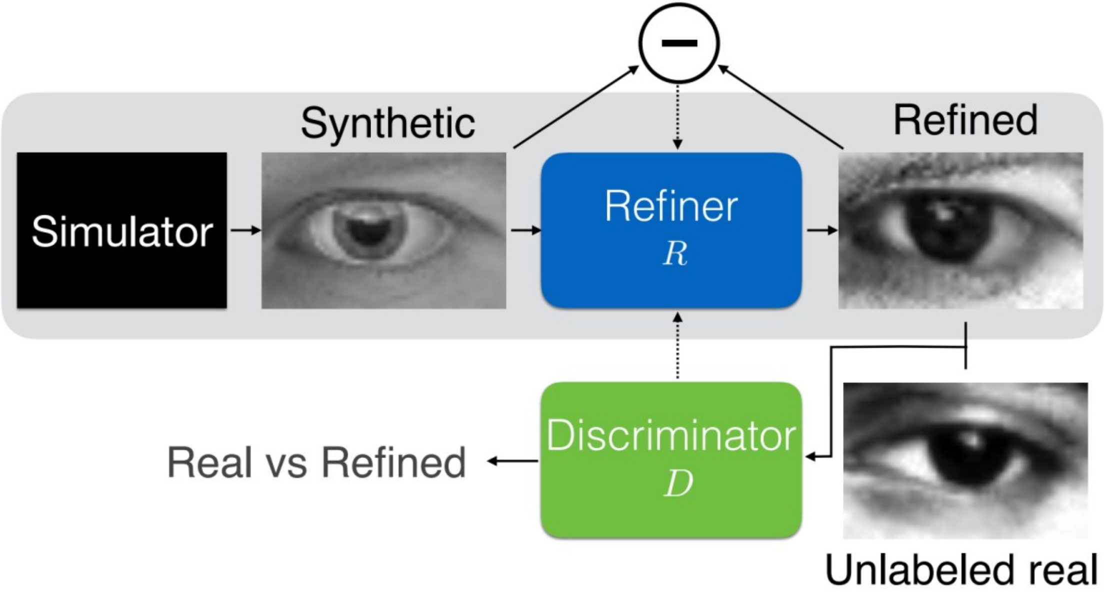
Sketch to image


Face manipulation

Super resolution

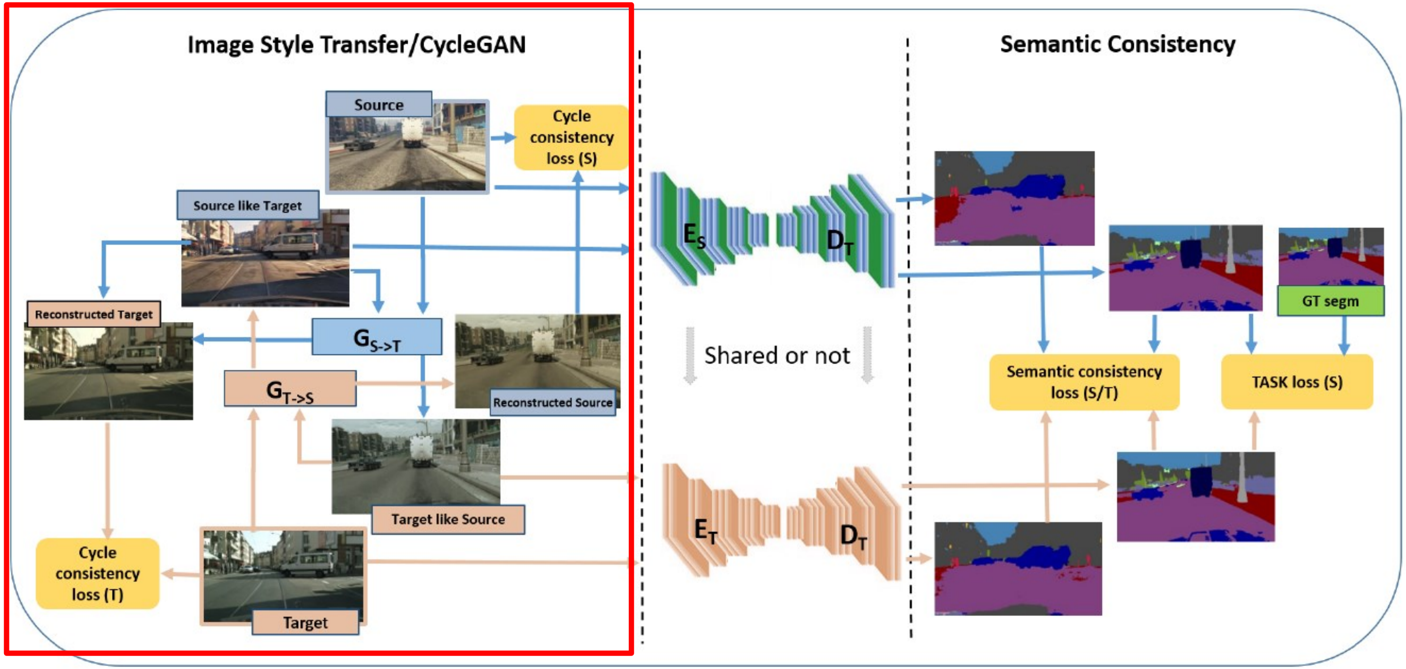
# Image to Image Translation as Image Style Transfer

## Synthetic to Real

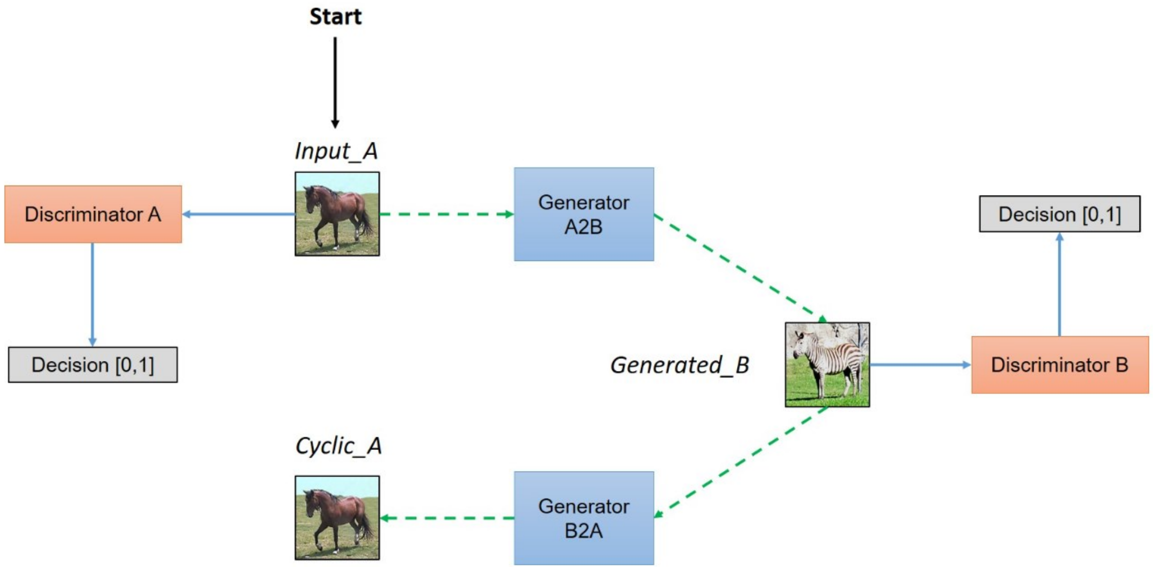



Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., & Webb, R. (2017). Learning from simulated and unsupervised images through adversarial training. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2107-2116).

# Domain Alignment: Image Level

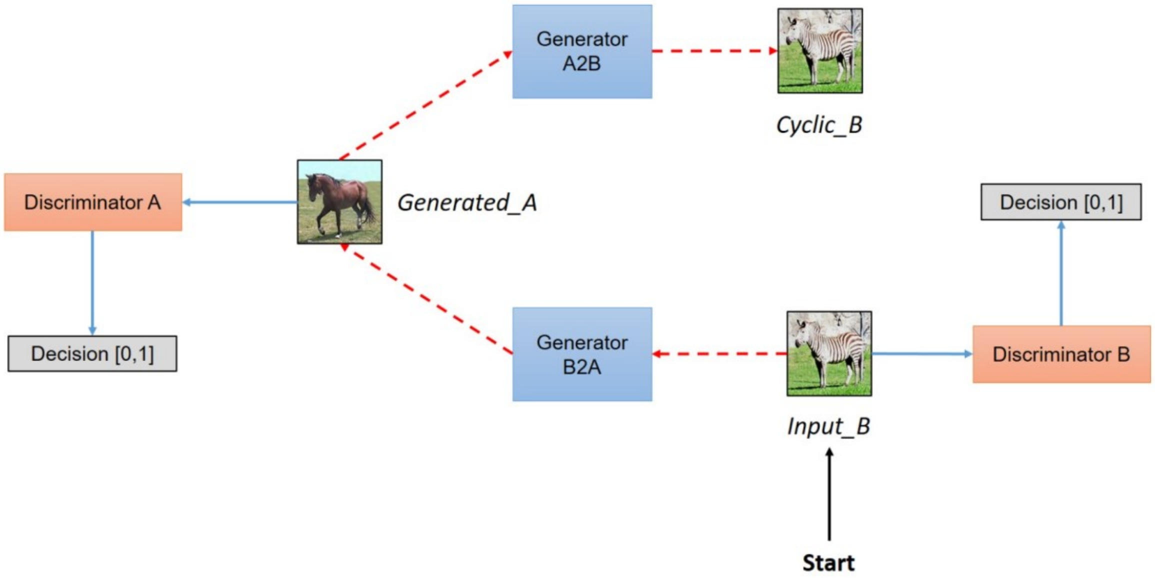


# Cycle-GAN




Jun-Yan Zhu*, Taesung Park*, Phillip Isola, and Alexei A. Efros. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", in IEEE International Conference on Computer Vision (ICCV), 2017. (* indicates equal contributions)

# Cycle-GAN



Generator A2B
Cyclic_B
Discriminator A
Generated_A
Decision [0,1]
Decision [0,1]
Generator B2A
Discriminator B
Input_B
Start

# Cycle-GAN

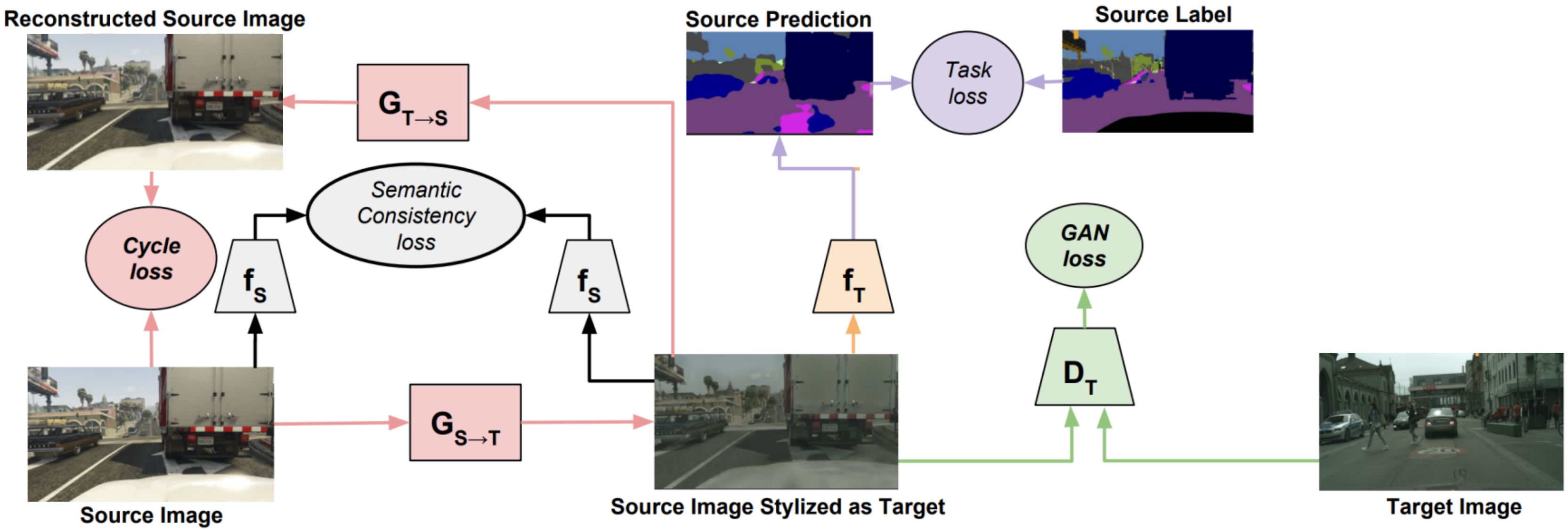
## Synthetic to Real



There are still artifacts due to the missing semantic information during the transformation process (e.g. sky becoming a tree).

# Cycada

## CycleGAN + Semantic Consistency





Hoffman, J., Tzeng, E., Park, T., Zhu, J. Y., Isola, P., Saenko, K., ... & Darrell, T. (2018, July). Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning* (pp. 1989-1998). PMLR.

# Cycada

Cityscapes

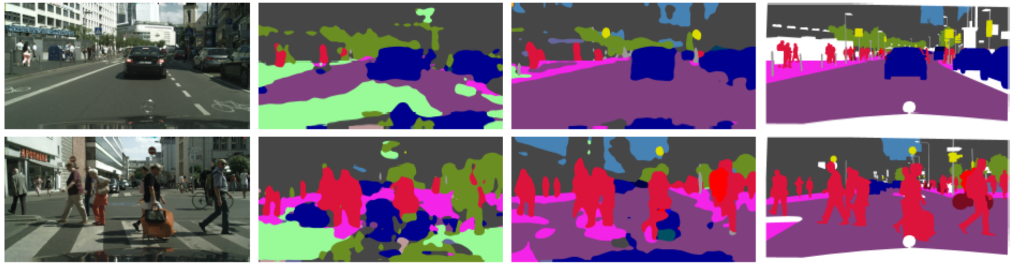
GTAV

GTAV to Cityscapes

# Cycada

## Some Results

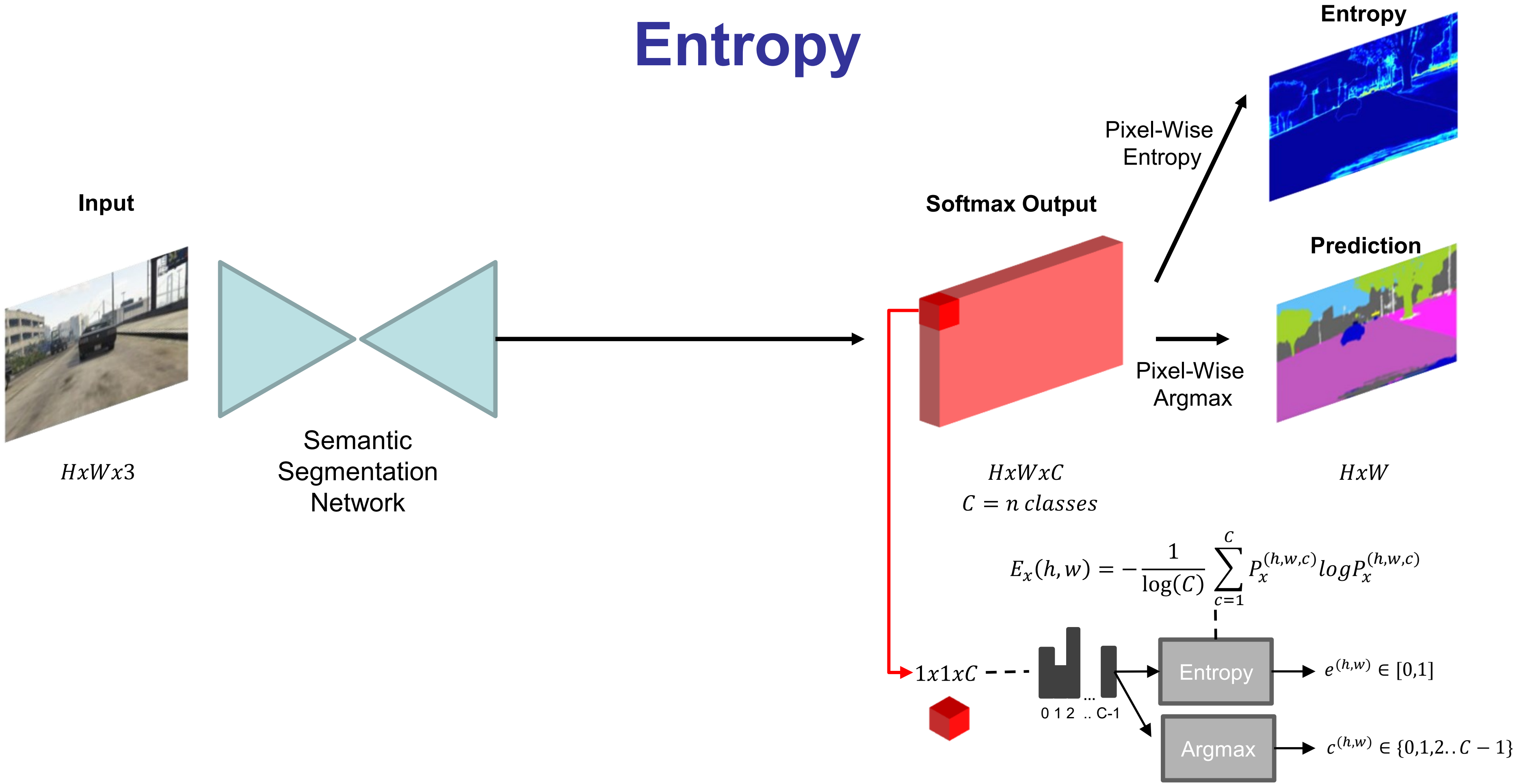
| GTA5 → Cityscapes | Architecture | road | sidewalk | building | wall | fence | pole | traffic light | traffic sign | vegetation | terrain | sky | person | rider | car | truck | bus | train | motorbike | bicycle | mIoU | fwIoU | Pixel acc. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Source only | A | 26.0 | 14.9 | 65.1 | 5.5 | 12.9 | 8.9 | 6.0 | 2.5 | 70.0 | 2.9 | 47.0 | 24.5 | 0.0 | 40.0 | 12.1 | 1.5 | 0.0 | 0.0 | 0.0 | 17.9 | 41.9 | 54.0 |
| FCNs in the wild* | A | 70.4 | 32.4 | 62.1 | 14.9 | 5.4 | 10.9 | 14.2 | 2.7 | 79.2 | 21.3 | 64.6 | 44.1 | 4.2 | 70.4 | 8.0 | 7.3 | 0.0 | 3.5 | 0.0 | 27.1 | — | — |
| CyCADA feat-only | A | **85.6** | 30.7 | 74.7 | 14.4 | 13.0 | 17.6 | 13.7 | 5.8 | 74.6 | 15.8 | **69.9** | 38.2 | 3.5 | 72.3 | 16.0 | 5.0 | 0.1 | 3.6 | 0.0 | 29.2 | 71.5 | 82.5 |
| CyCADA pixel-only | A | 83.5 | **38.3** | **76.4** | 20.6 | **16.5** | 22.2 | **26.2** | **21.9** | **80.4** | 28.7 | 65.7 | 49.4 | 4.2 | 74.6 | 16.0 | 26.6 | 2.0 | 8.0 | 0.0 | 34.8 | 73.1 | 82.8 |
| CyCADA pixel+feat | A | 85.2 | 37.2 | **76.5** | **21.8** | 15.0 | **23.8** | 22.9 | 21.5 | **80.5** | **31.3** | 60.7 | **50.5** | **9.0** | **76.9** | **17.1** | **28.2** | **4.5** | **9.8** | 0.0 | **35.4** | **73.8** | **83.6** |
| Oracle - Target Super | A | 96.4 | 74.5 | 87.1 | 35.3 | 37.8 | 36.4 | 46.9 | 60.1 | 89.0 | 54.3 | 89.8 | 65.6 | 35.9 | 89.4 | 38.6 | 64.1 | 38.6 | 40.5 | 65.1 | 60.3 | 87.6 | 93.1 |
| Source only | B | 42.7 | 26.3 | 51.7 | 5.5 | 6.8 | 13.8 | 23.6 | 6.9 | 75.5 | 11.5 | 36.8 | 49.3 | 0.9 | 46.7 | 3.4 | 5.0 | 0.0 | 5.0 | 1.4 | 21.7 | 47.4 | 62.5 |
| CyCADA feat-only | B | 78.1 | 31.1 | 71.2 | 10.3 | 14.1 | 29.8 | 28.1 | 20.9 | 74.0 | 16.8 | 51.9 | 53.6 | 6.1 | 65.4 | 8.2 | 20.9 | 1.8 | 13.9 | 5.9 | 31.7 | 67.4 | 78.4 |
| CyCADA pixel-only | B | 63.7 | 24.7 | 69.3 | 21.2 | 17.0 | 30.3 | 33.0 | **32.0** | 80.5 | 25.3 | 62.3 | 62.0 | **15.1** | 73.1 | 19.8 | 23.6 | 5.5 | 16.2 | **28.7** | 37.0 | 63.8 | 75.4 |
| CyCADA pixel+feat | B | **79.1** | **33.1** | **77.9** | **23.4** | **17.3** | **32.1** | **33.3** | 31.8 | **81.5** | **26.7** | **69.0** | **62.8** | 14.7 | **74.5** | **20.9** | **25.6** | **6.9** | **18.8** | 20.4 | **39.5** | **72.4** | **82.3** |
| Oracle - Target Super | B | 97.3 | 79.8 | 88.6 | 32.5 | 48.2 | 56.3 | 63.6 | 73.3 | 89.0 | 58.9 | 93.0 | 78.2 | 55.2 | 92.2 | 45.0 | 67.3 | 39.6 | 49.9 | 73.6 | 67.4 | 89.6 | 94.3 |



(a) Test Image (b) Source Prediction (c) CyCADA Prediction (d) Ground Truth

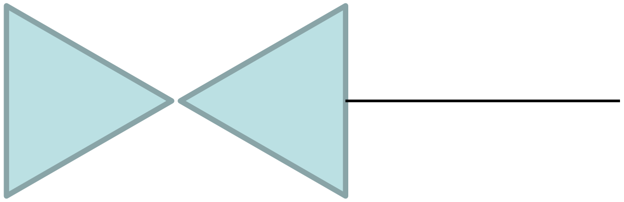
# Entropy Minimization

# Entropy

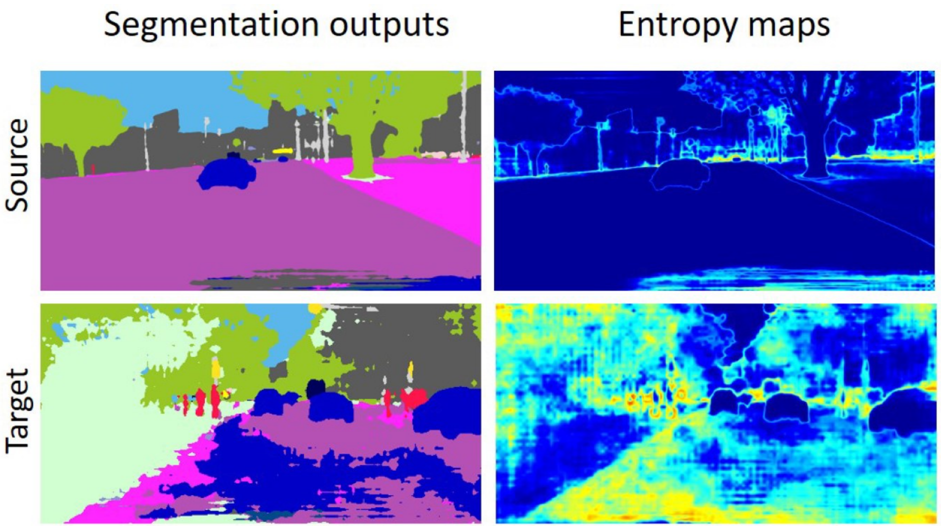


Input
$HxWx3$
Semantic Segmentation Network
Softmax Output
$HxWxC$
$C = n\ classes$
Pixel-Wise Entropy
Entropy
Pixel-Wise Argmax
Prediction
$HxW$
$E_x(h,w) = -\frac{1}{\log(C)} \sum_{c=1}^{C} P_x^{(h,w,c)} log P_x^{(h,w,c)}$
$1x1xC$
0 1 2 .. C-1
Entropy
Argmax
$e^{(h,w)} \in [0,1]$
$c^{(h,w)} \in \{0,1,2..C-1\}$

# Source vs Target Entropy

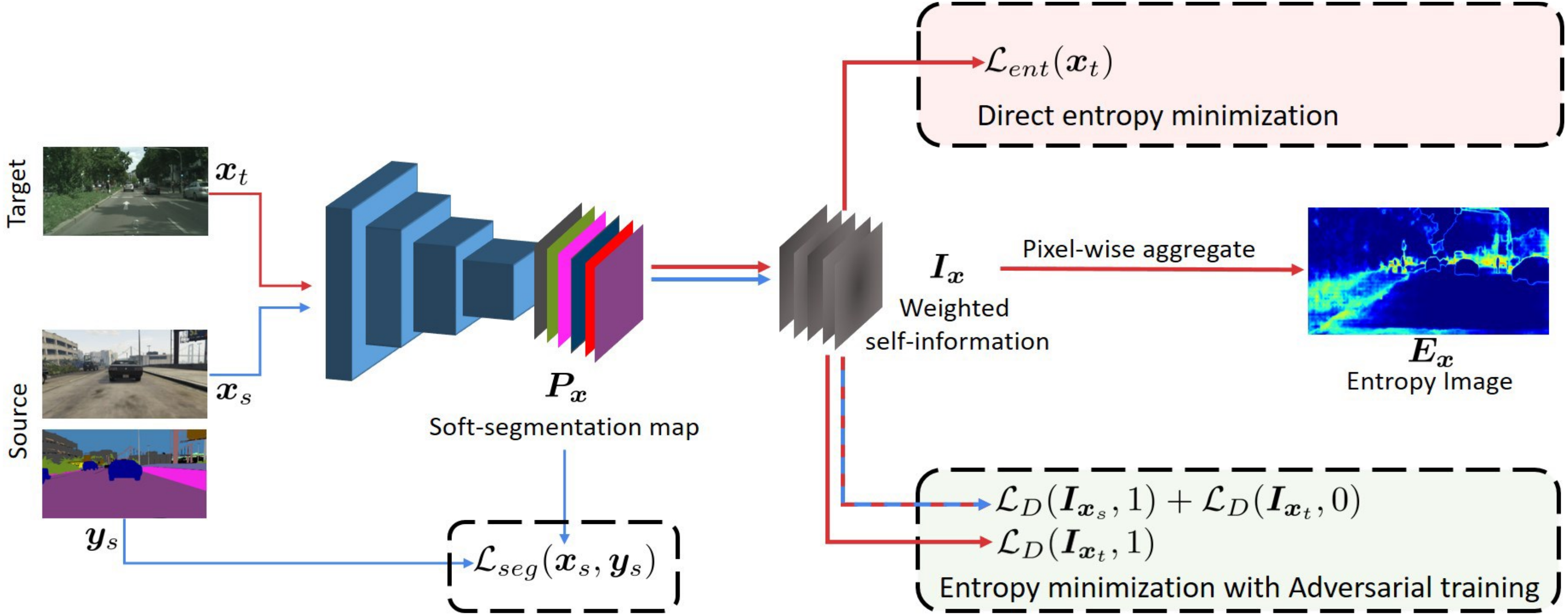


Semantic Segmentation Network
**Trained on Source Domain**




Models trained only on source domain tend to produce *over-confident*, *i.e.*, low-entropy, predictions on source-like images and *under-confident*, *i.e.*, high-entropy, predictions on target-like ones

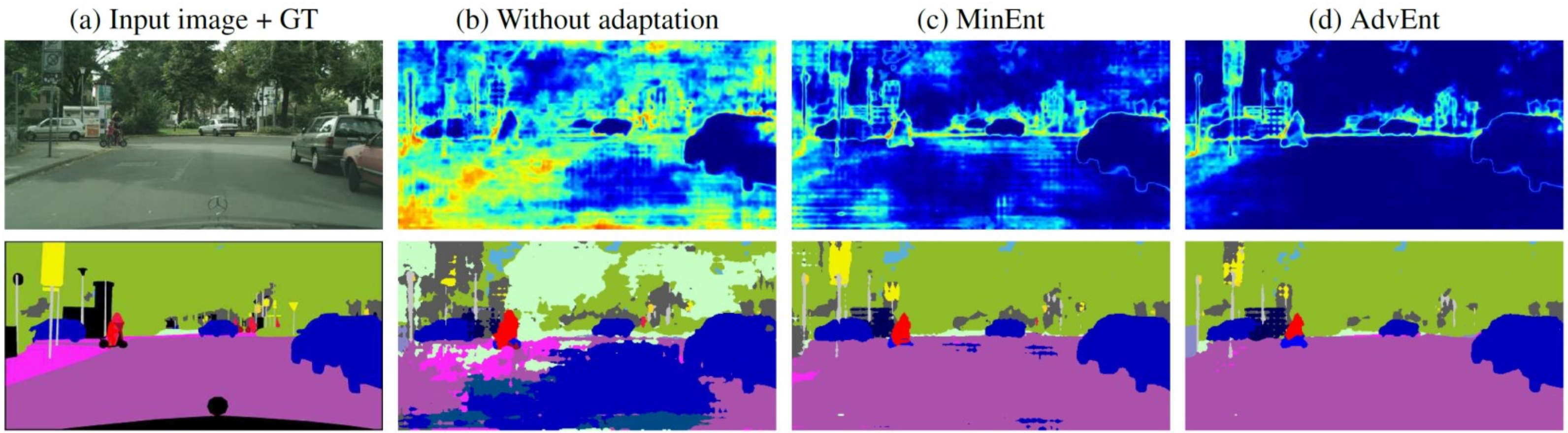
# Source vs Target Entropy




Vu, T. H., Jain, H., Bucher, M., Cord, M., & Pérez, P. (2019). Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 2517-2526).

# Ablation Study

MinEnt manage to produce correct predictions at high level of confidence.




Model trained only on source supervision produces noisy segmentation predictions as well as high entropy activations, with a few exceptions on some classes like “building” and “car”.
Still, there exist many confident predictions (low entropy) which are completely wrong.

AdvEnt achieves lower prediction entropy compared to the MinEnt model.

# Quantitative Results

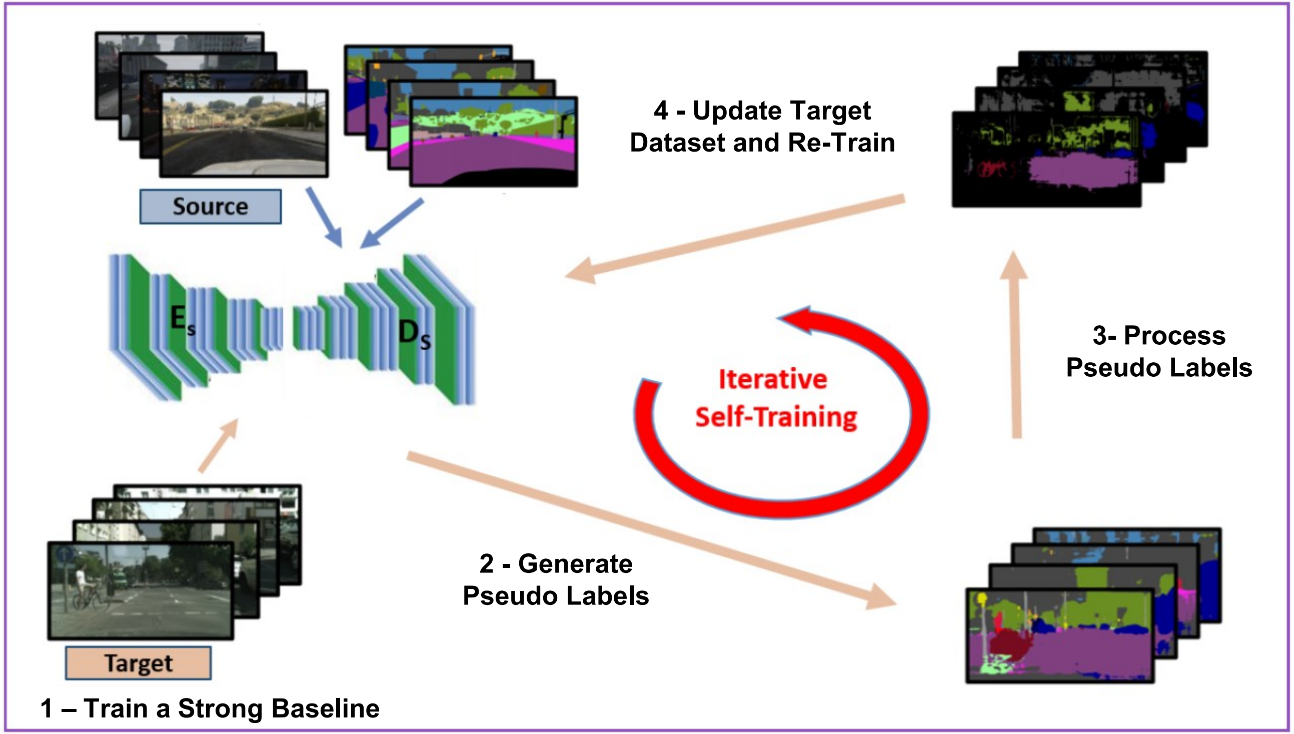
| | Models | Appr. | road | sidewalk | building | wall | fence | pole | light | sign | veg | terrain | sky | person | rider | car | truck | bus | train | mbike | bike | mIoU |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Deeplab-V2 with VGG-16 | FCNs in the Wild [15] | Adv | 70.4 | 32.4 | 62.1 | 14.9 | 5.4 | 10.9 | 14.2 | 2.7 | 79.2 | 21.3 | 64.6 | 44.1 | 4.2 | 70.4 | 8.0 | 7.3 | 0.0 | 3.5 | 0.0 | 27.1 |
| | CyCADA [14] | Adv | 83.5 | **38.3** | 76.4 | 20.6 | 16.5 | 22.2 | 26.2 | 21.9 | 80.4 | 28.7 | 65.7 | 49.4 | 4.2 | 74.6 | 16.0 | 26.6 | 2.0 | 8.0 | 0.0 | 34.8 |
| | Adapt-SegMap [41] | Adv | 87.3 | 29.8 | 78.6 | 21.1 | 18.2 | 22.5 | 21.5 | 11.0 | 79.7 | 29.6 | 71.3 | 46.8 | 6.5 | 80.1 | 23.0 | 26.9 | 0.0 | 10.6 | 0.3 | 35.0 |
| | Self-Training [51] | ST | 83.8 | 17.4 | 72.1 | 14.3 | 2.9 | 16.5 | 16.0 | 6.8 | 81.4 | 24.2 | 47.2 | 40.7 | 7.6 | 71.7 | 10.2 | 7.6 | 0.5 | 11.1 | 0.9 | 28.1 |
| | Self-Training + CB [51] | ST | 66.7 | 26.8 | 73.7 | 14.8 | 9.5 | 28.3 | 25.9 | 10.1 | 75.5 | 15.7 | 51.6 | 47.2 | 6.2 | 71.9 | 3.7 | 2.2 | 5.4 | 18.9 | 32.4 | 30.9 |
| | Ours (MinEnt) | Ent | 85.1 | 18.9 | 76.3 | 32.4 | 19.7 | 19.9 | 21.0 | 8.9 | 76.3 | 26.2 | 63.1 | 42.8 | 5.9 | 80.8 | 20.2 | 9.8 | 0.0 | 14.8 | 0.6 | 32.8 |
| | Ours (AdvEnt) | Adv | 86.9 | 28.7 | 78.7 | 28.5 | 25.2 | 17.1 | 20.3 | 10.9 | 80.0 | 26.4 | 70.2 | 47.1 | 8.4 | 81.5 | 26.0 | 17.2 | **18.9** | 11.7 | 1.6 | 36.1 |
| Deeplab-V2 with ResNet 101 | Adapt-SegMap [41] | Adv | 86.5 | 36.0 | 79.9 | 23.4 | 23.3 | 23.9 | 35.2 | 14.8 | 83.4 | 33.3 | 75.6 | 58.5 | 27.6 | 73.7 | 32.5 | 35.4 | 3.9 | 30.1 | 28.1 | 42.4 |
| | Adapt-SegMap* | Adv | 85.5 | 18.4 | 80.8 | 29.1 | 24.6 | 27.9 | 33.1 | 20.9 | 83.8 | 31.2 | 75.0 | 57.5 | 28.6 | 77.3 | 32.3 | 30.9 | 1.1 | 28.7 | 35.9 | 42.2 |
| | Ours (MinEnt) | Ent | 84.4 | 18.7 | 80.6 | 23.8 | 23.2 | 28.4 | **36.9** | 23.4 | 83.2 | 25.2 | **79.4** | 59.0 | 29.9 | 78.5 | 33.7 | 29.6 | 1.7 | 29.9 | 33.6 | 42.3 |
| | Ours (MinEnt + ER) | Ent | 84.2 | 25.2 | 77.0 | 17.0 | 23.3 | 24.2 | 33.3 | **26.4** | 80.7 | 32.1 | 78.7 | 57.5 | 30.0 | 77.0 | 37.9 | 44.3 | 1.8 | 31.4 | **36.9** | 43.1 |
| | Ours (AdvEnt) | Adv | **89.9** | 36.5 | **81.6** | **29.2** | 25.2 | **28.5** | 32.3 | 22.4 | 83.9 | 34.0 | 77.1 | 57.4 | 27.9 | 83.7 | 29.4 | 39.1 | 1.5 | 28.4 | 23.3 | 43.8 |
| | Ours (AdvEnt+MinEnt) | A+E | 89.4 | 33.1 | 81.0 | 26.6 | **26.8** | 27.2 | 33.5 | 24.7 | **83.9** | **36.7** | 78.8 | **58.7** | **30.5** | **84.8** | **38.5** | **44.5** | 1.7 | **31.6** | 32.4 | **45.5** |

# Self-Training

S. Fralick, "Learning to recognize patterns without a teacher," in *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 57-64, January 1967, doi: 10.1109/TIT.1967.1053952.

# Self-Training

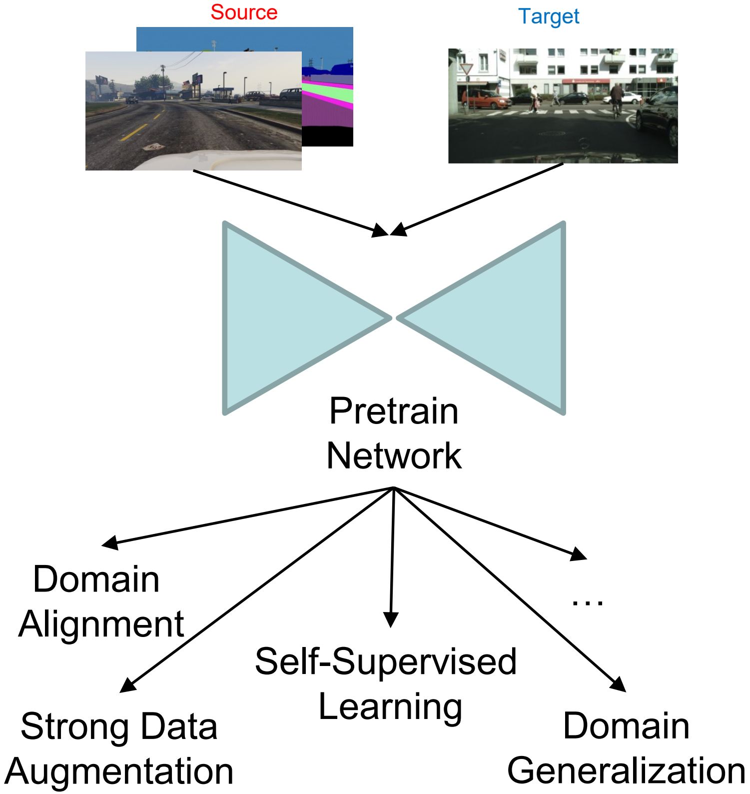
## Overview




Zou, Y., Yu, Z., Kumar, B. V. K., & Wang, J. (2018). Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 289-305).

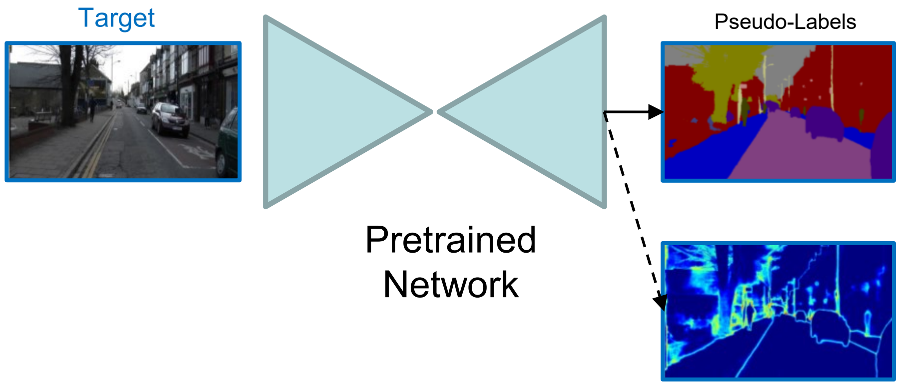
# Self-Training

## 1 – Train a Strong Baseline



# Self-Training

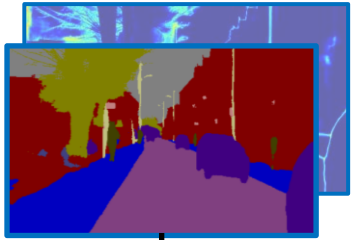
**2 – Generate Pseudo Labels**




[OPTIONAL]
Auxiliary information used to filter wrong proxies such as **Entropy**, Softmax, Logits, etc.

# Self-Training

## 3 – Process Pseudo Labels



Processing

**Filtering**

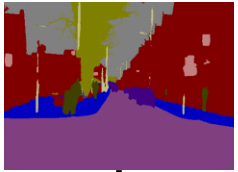
Refinement

…

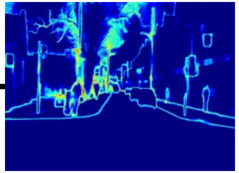
Cut&Paste

## Example Naïve Filtering

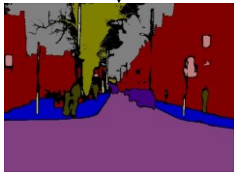
Pseudo-Label



Filtering Low-Confident



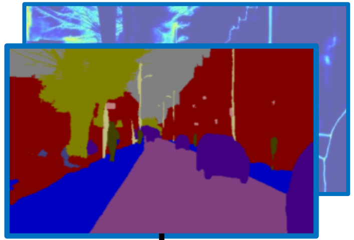
Confidence Information (e.g., Entropy, Softmax, ..)



Filtered Pseudo-Label

# Self-Training

## 3 – Process Pseudo Labels



Processing

**Filtering**

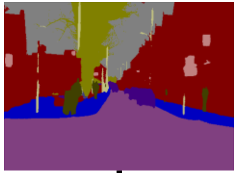
Refinement

…

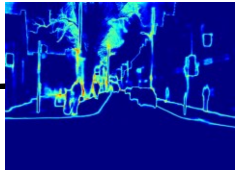
Cut&Paste

## Class-Balanced Filtering

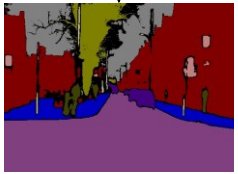
Pseudo-Label



**Class-Balanced** Filtering Low-Confident



Confidence Information (e.g., Entropy, Softmax, ..)



Filtered Pseudo-Label

Zou, Y., Yu, Z., Kumar, B. V. K., & Wang, J. (2018). Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 289-305).

# Class-Balanced Self-Training (CBST)

## Class-Balanced Filtering

Softmax Output
for each Target Image

1- Re-arrange softmax maps as pixels arrays for each class (argmax of softmax)

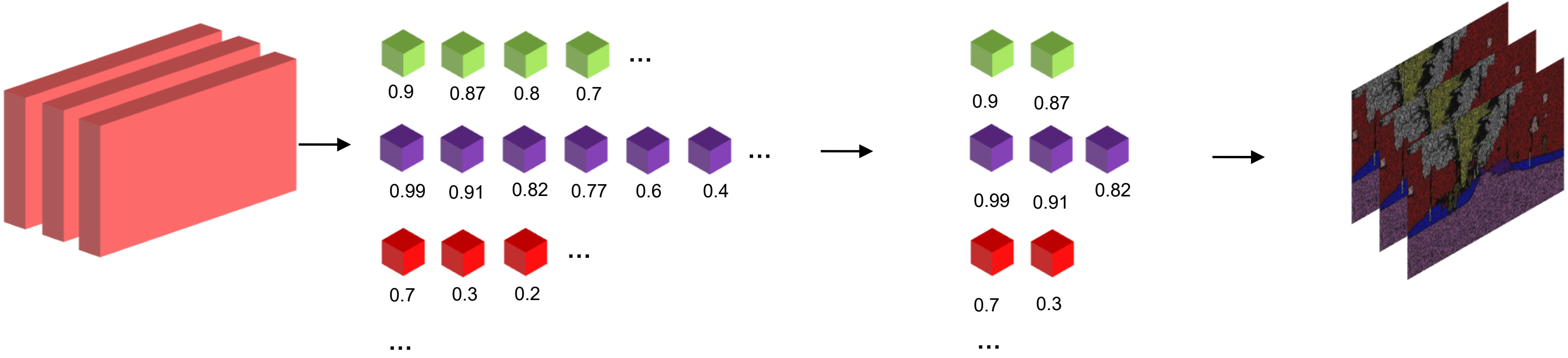
2- Sort from most to least confident (Softmax maximum for each pixel as confidence)

Select $p_c$ most confident pixels of each class

$p_c = p \in [0 - 100]\%$ of pixels of class $c$

$p$ is the same for all classes

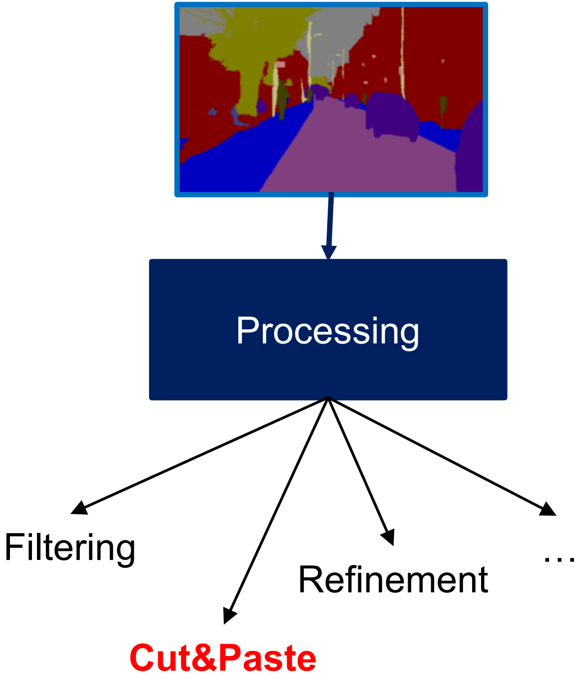
Filtered
Pseudo Labels



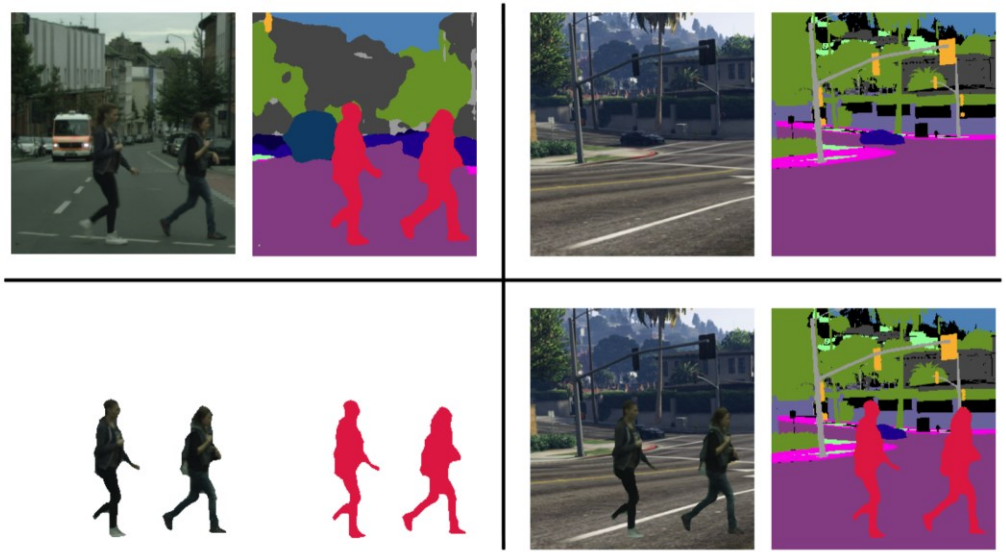

Zou, Y., Yu, Z., Kumar, B. V. K., & Wang, J. (2018). Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 289-305).

# Self-Training

## 3 – Process Pseudo Labels



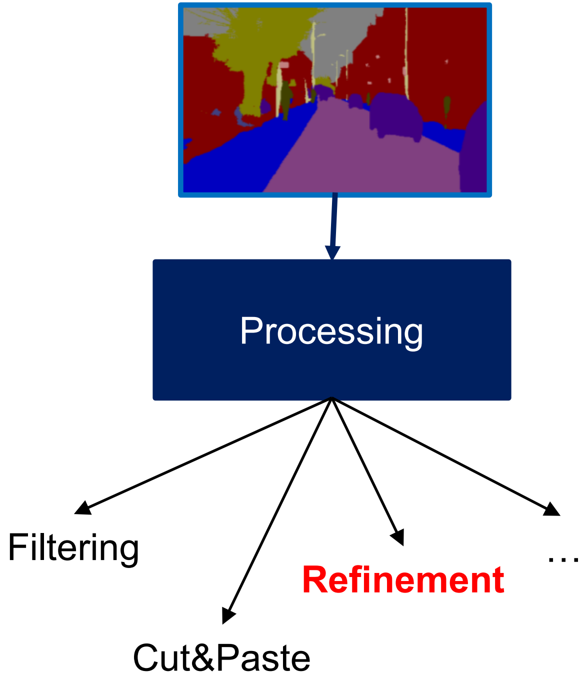

## Example of Cut&Paste



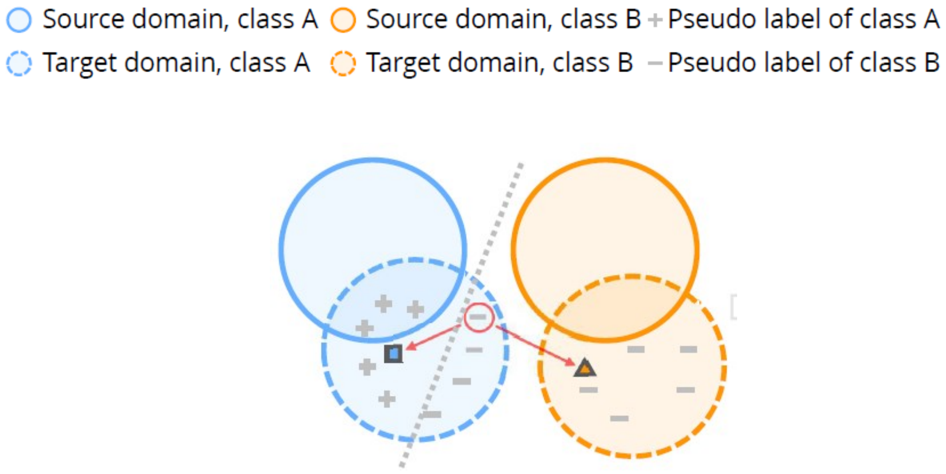
Cardace, A., Ramirez, P. Z., Salti, S., & Di Stefano, L. (2022). Shallow Features Guide Unsupervised Domain Adaptation for Semantic Segmentation at Class Boundaries. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 1160-1170).

# Self-Training

## 3 – Process Pseudo Labels




The pseudo labels are obtained according to a strict confidence threshold, **while high scores are not necessarily correct,** making the network fail to learn reliable knowledge in the target domain.

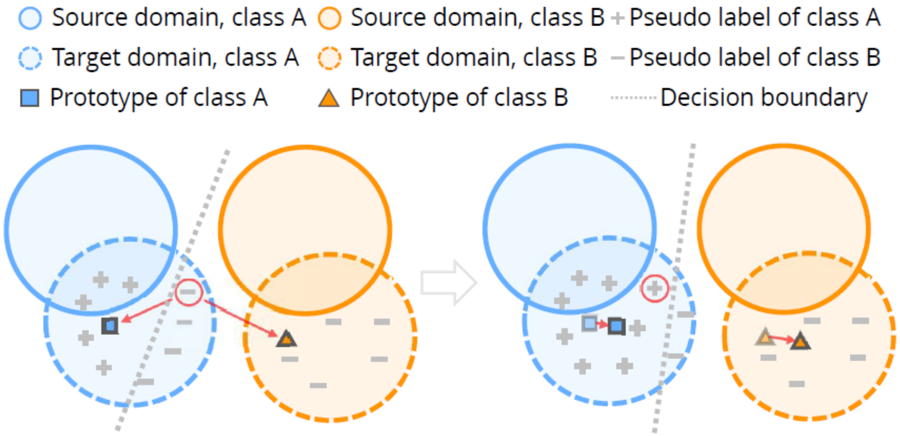



The decision boundary (dashed line) crosses the distribution of the target data and induces incorrect pseudo label predictions. This is because the network is unaware of the target distribution when generating pseudo labels.

# Self-Training

## Refinement with Prototypes

**ProDA**




Calculate the prototypes of each class **on-the-fly** and rely on these prototypes to online **correct** the false pseudo labels.

Each pixel softmax output $p_t$ is multiplied by weights $w_t$ accordingly to distances w.r.t. prototypes $\eta$ before doing argmax $\varepsilon$. $\tilde{f}$ is a mean teacher.

$$\hat{y}_t^{(i,j)} = \varepsilon\left(w_t^{(i,k)} p_t^{(i,k)}\right)$$

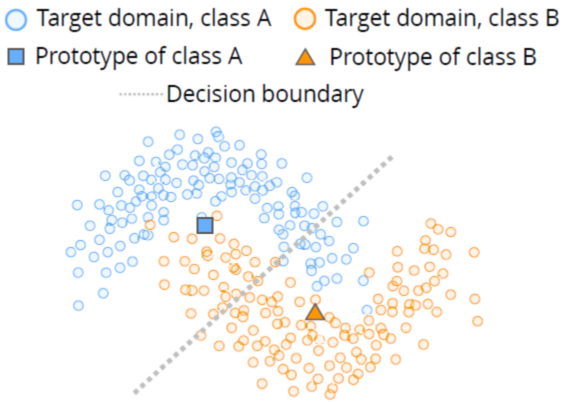
$$w_t^{(i,k)} = \frac{\exp\left(-\frac{\left\|\tilde{f}(x_t)^i - \eta^{(k)}\right\|}{\tau}\right)}{\sum_{k'} \exp\left(-\frac{\left\|\tilde{f}(x_t)^i - \eta^{(k')}\right\|}{\tau}\right)}$$

Prototypes are estimated after each iteration as a moving average of the cluster centroids in mini-batch

Zhang, P., Zhang, B., Zhang, T., Chen, D., Wang, Y., & Wen, F. (2021). Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 12414-12424).

# Self-Training
## Refinement with Prototypes

The network may induce
dispersed feature distribution in the target domain which is
hardly differentiated by a linear classifier.



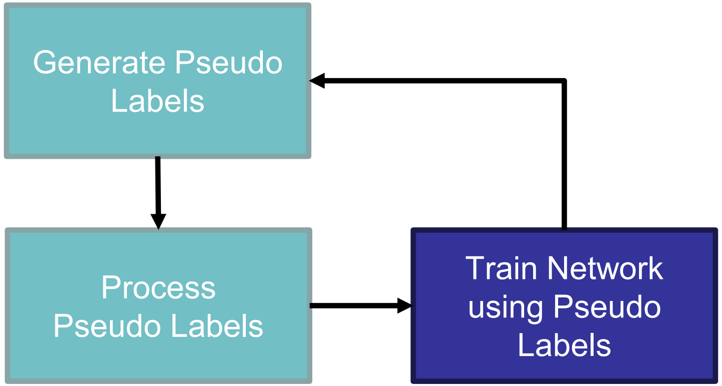

In this case, the prototypes fail to rectify the labels of the data whose features lie in the far end of the cluster even when the target features from the source model are well-separated.

**Idea:** Forcing clustering by constraining two augmented version of the same data $x_t$ to have the same distance w.r.t. class prototypes.

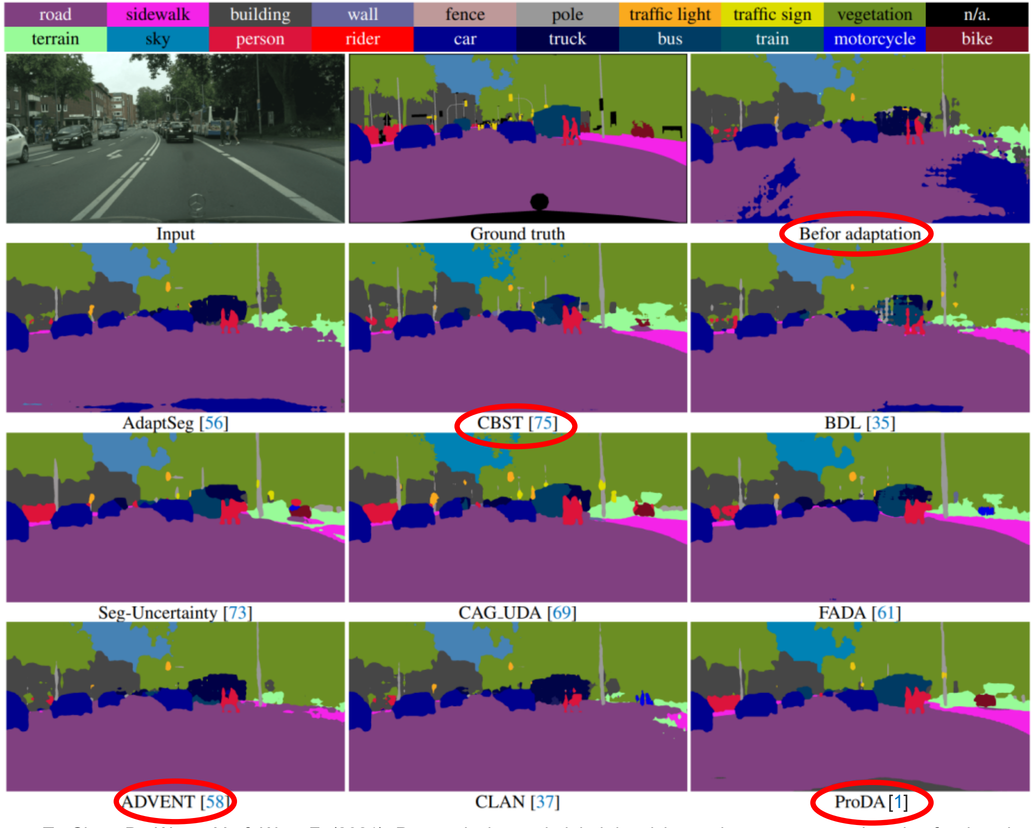

Zhang, P., Zhang, B., Zhang, T., Chen, D., Wang, Y., & Wen, F. (2021). Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 12414-12424).

# Self-Training

**4 – Iterative Process**



# Some Qualitative Results




[1] Zhang, P., Zhang, B., Zhang, T., Chen, D., Wang, Y., & Wen, F. (2021). Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 12414-12424).

# Some Quantitative Results

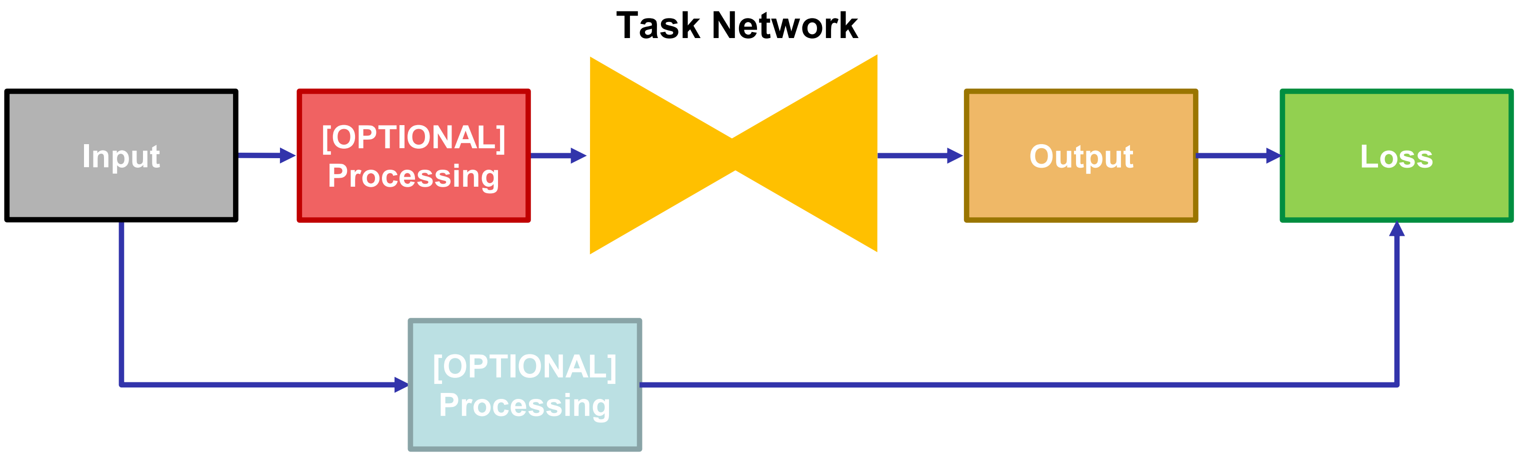
| | road | sideway | building | wall | fence | pole | light | sign | vege. | terrace | sky | person | rider | car | truck | bus | train | motor | bike | mIoU | gain |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Source | 75.8 | 16.8 | 77.2 | 12.5 | 21.0 | 25.5 | 30.1 | 20.1 | 81.3 | 24.6 | 70.3 | 53.8 | 26.4 | 49.9 | 17.2 | 25.9 | 6.5 | 25.3 | 36.0 | 36.6 | +0.0 |
| AdaptSeg [55] | 86.5 | 25.9 | 79.8 | 22.1 | 20.0 | 23.6 | 33.1 | 21.8 | 81.8 | 25.9 | 75.9 | 57.3 | 26.2 | 76.3 | 29.8 | 32.1 | 7.2 | 29.5 | 32.5 | 41.4 | +4.8 |
| CyCADA [27] | 86.7 | 35.6 | 80.1 | 19.8 | 17.5 | 38.0 | 39.9 | 41.5 | 82.7 | 27.9 | 73.6 | 64.9 | 19.0 | 65.0 | 12.0 | 28.6 | 4.5 | 31.1 | 42.0 | 42.7 | +6.1 |
| CLAN [37] | 87.0 | 27.1 | 79.6 | 27.3 | 23.3 | 28.3 | 35.5 | 24.2 | 83.6 | 27.4 | 74.2 | 58.6 | 28.0 | 76.2 | 33.1 | 36.7 | 6.7 | 31.9 | 31.4 | 43.2 | +6.6 |
| APODA [68] | 85.6 | 32.8 | 79.0 | 29.5 | 25.5 | 26.8 | 34.6 | 19.9 | 83.7 | 40.6 | 77.9 | 59.2 | 28.3 | 84.6 | 34.6 | 49.2 | 8.0 | 32.6 | 39.6 | 45.9 | +9.3 |
| PatchAlign [57] | **92.3** | 51.9 | 82.1 | 29.2 | 25.1 | 24.5 | 33.8 | 33.0 | 82.4 | 32.8 | 82.2 | 58.6 | 27.2 | 84.3 | 33.4 | 46.3 | 2.2 | 29.5 | 32.3 | 46.5 | +9.9 |
| ADVENT [58] | 89.4 | 33.1 | 81.0 | 26.6 | 26.8 | 27.2 | 33.5 | 24.7 | 83.9 | 36.7 | 78.8 | 58.7 | 30.5 | 84.8 | 38.5 | 44.5 | 1.7 | 31.6 | 32.4 | 45.5 | +8.9 |
| BDL [35] | 91.0 | 44.7 | 84.2 | 34.6 | 27.6 | 30.2 | 36.0 | 36.0 | 85.0 | 43.6 | 83.0 | 58.6 | 31.6 | 83.3 | 35.3 | 49.7 | 3.3 | 28.8 | 35.6 | 48.5 | +11.9 |
| FADA [61] | 91.0 | 50.6 | **86.0** | 43.4 | 29.8 | 36.8 | 43.4 | 25.0 | 86.8 | 38.3 | **87.4** | 64.0 | 38.0 | 85.2 | 31.6 | 46.1 | 6.5 | 25.4 | 37.1 | 50.1 | +13.5 |
| CBST [75] | 91.8 | 53.5 | 80.5 | 32.7 | 21.0 | 34.0 | 28.9 | 20.4 | 83.9 | 34.2 | 80.9 | 53.1 | 24.0 | 82.7 | 30.3 | 35.9 | 16.0 | 25.9 | 42.8 | 45.9 | +9.3 |
| MRKLD [76] | 91.0 | 55.4 | 80.0 | 33.7 | 21.4 | 37.3 | 32.9 | 24.5 | 85.0 | 34.1 | 80.8 | 57.7 | 24.6 | 84.1 | 27.8 | 30.1 | 26.9 | 26.0 | 42.3 | 47.1 | +10.5 |
| CAG_UDA [69] | 90.4 | 51.6 | 83.8 | 34.2 | 27.8 | 38.4 | 25.3 | 48.4 | 85.4 | 38.2 | 78.1 | 58.6 | 34.6 | 84.7 | 21.9 | 42.7 | **41.1** | 29.3 | 37.2 | 50.2 | +13.6 |
| Seg-Uncertainty [73] | 90.4 | 31.2 | 85.1 | 36.9 | 25.6 | 37.5 | 48.8 | 48.5 | 85.3 | 34.8 | 81.1 | 64.4 | 36.8 | 86.3 | 34.9 | 52.2 | 1.7 | 29.0 | 44.6 | 50.3 | +13.7 |
| *ProDA* | 87.8 | **56.0** | 79.7 | **46.3** | **44.8** | **45.6** | **53.5** | **53.5** | **88.6** | **45.2** | 82.1 | **70.7** | **39.2** | **88.8** | **45.5** | **59.4** | 1.0 | **48.9** | **56.4** | **57.5** | **+20.9** |
| Oracle | | | | | | | | | | | | | | | | | | | | 65.1 | |

[1]

[1] Zhang, P., Zhang, B., Zhang, T., Chen, D., Wang, Y., & Wen, F. (2021). Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 12414-12424).

# Self-supervised learning

# Self-supervised Tasks



# Examples of Self-Supervised Tasks on Images

**Image Colorization**

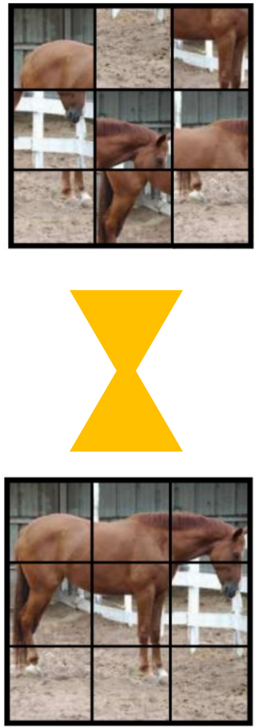
**Image Rotation**

**Auto-Encoder**

# Examples of Self-Supervised Tasks on Images

Denoising Auto-Encoder

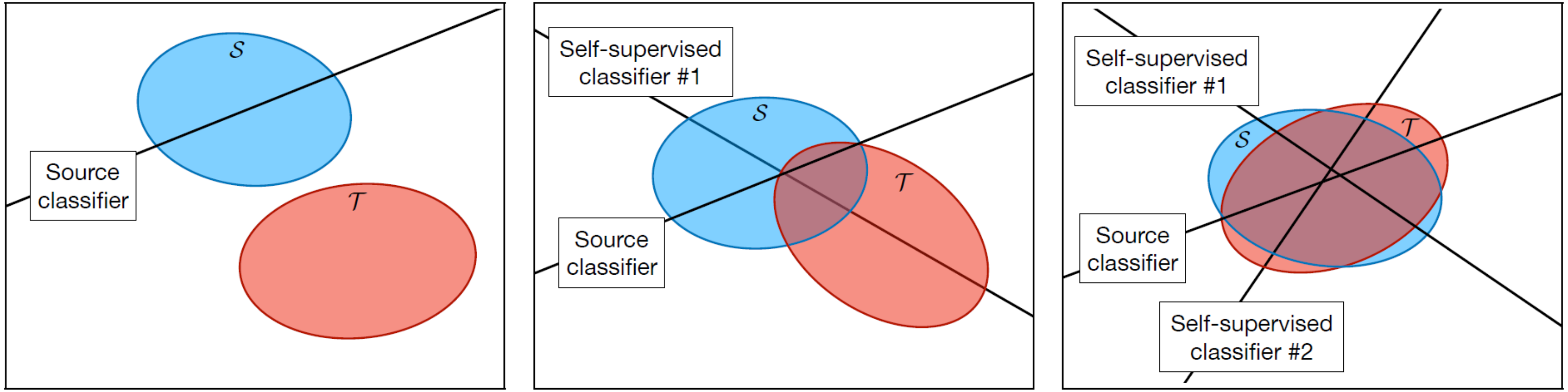
Masked Auto-Encoders



Jigsaw Puzzle

# Self-supervised learning in UDA

## Self-Supervised Tasks as Auxiliary Tasks for Domain Alignment




Source domain is far away from the target domain, and a source classifier cannot generalize to the target.

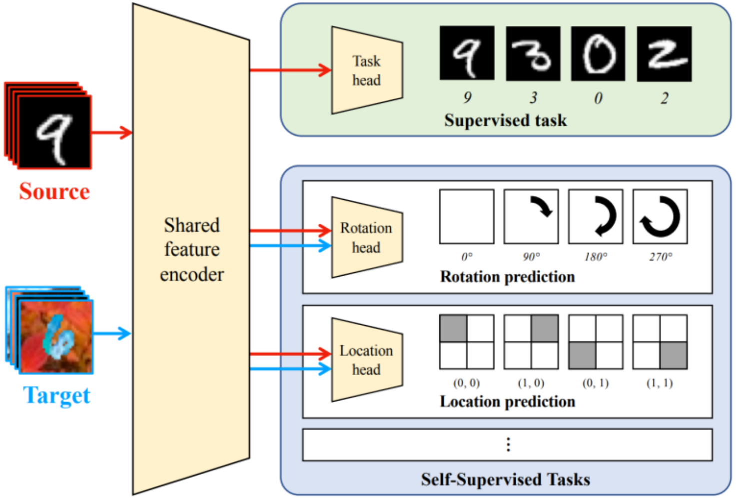
Training a shared representation to support one self-supervised task on both domains can align the source and target along one direction.

Using multiple self-supervised tasks can further align the domains along multiple directions.


Sun, Y., Tzeng, E., Darrell, T., & Efros, A. A. (2019). Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*.

# Self-supervised learning in UDA

## Self-Supervised Tasks as Auxiliary Tasks




**Select and correctly using the Auxiliary Tasks is difficult:**

- It should help reasoning about the Target Task
- It should be “aligned” across domains (e.g., should not require capturing information on the factors where the domains are meaningless different)


Sun, Y., Tzeng, E., Darrell, T., & Efros, A. A. (2019). Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*.

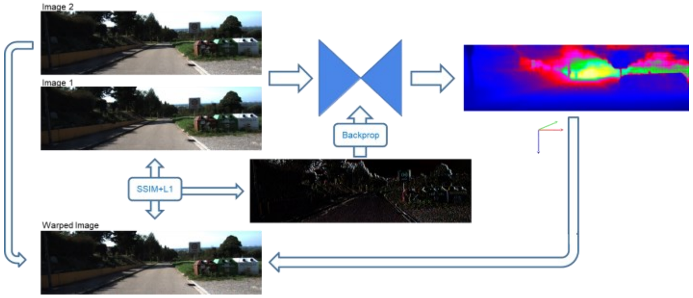
# Self-supervised learning in UDA
## Self-Supervised Tasks as Auxiliary Tasks

**Example: Colorization**

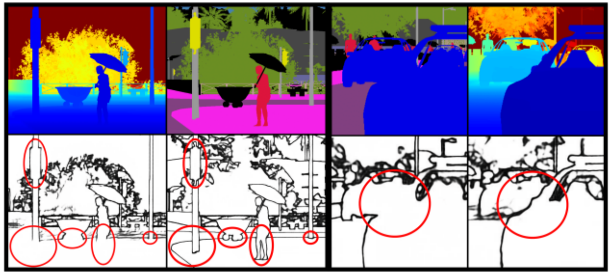
If used only on the target domain may help discriminability, i.e., reasoning how to color an image is connected to the object semantic. However, if performed on both source and target domain would make feature focusing on color, increasing the domain-gap.

Sun, Y., Tzeng, E., Darrell, T., & Efros, A. A. (2019). Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*.

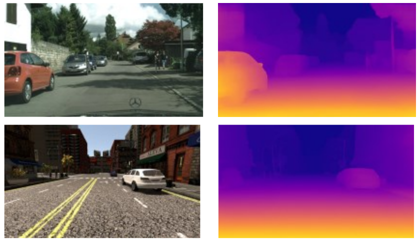
# Some Reasons Why Depth can be a good Auxiliary Task for Semantic Segmentation



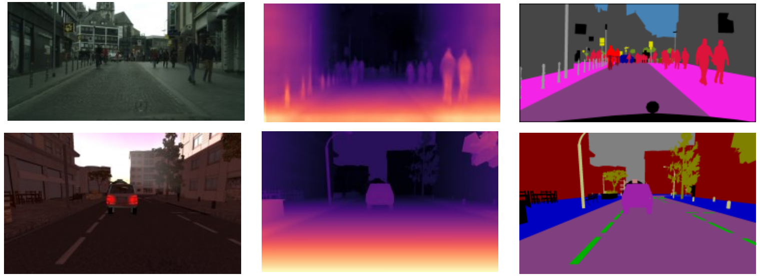

Depth can be addressed in a self-supervised manner



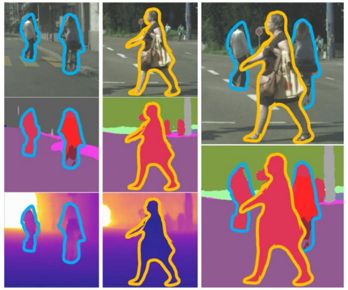
Depth and semantic share similar edge structure.



Depth Structures are Similar Across Domains



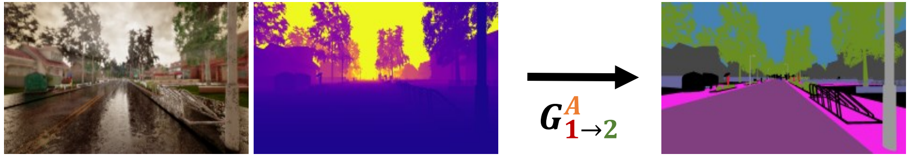
Correlations between tasks are moderately domain-invariant (e.g., road flat, sky far away).



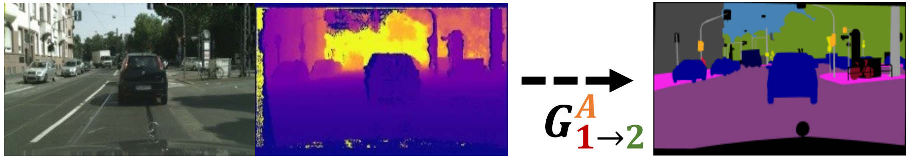
Depth information can be useful for some geometric data augmentation

# Self-supervised Learning in UDA for Semantic Segmentation

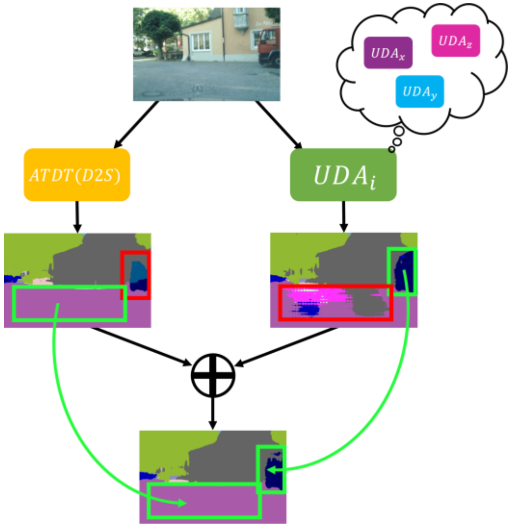
## Self-Supervised Depth




Learning relationships between tasks in Source Domain with labels




Relationship generalize well across domains and can be used to extract information from the depth information

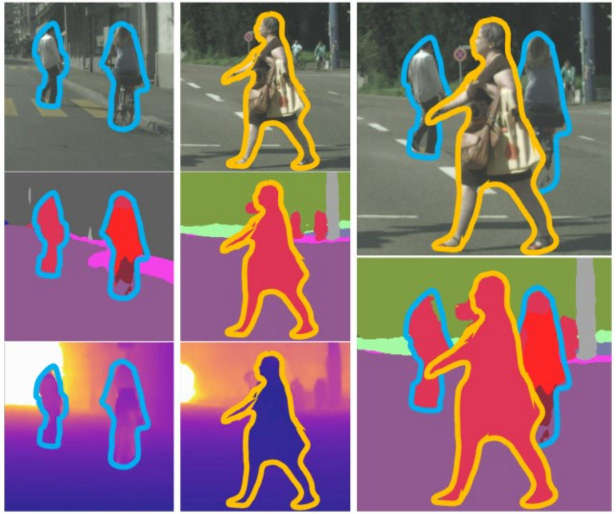



Semantic from depth is strong in areas with domain-invariant across tasks relationship.
(e.g. sky is far and in top image regions).
Merge with any other standard UDA method.


Cardace, A., De Luigi, L., Ramirez, P. Z., Salti, S., & Di Stefano, L. (2022). Plugging Self-Supervised Monocular Depth into Unsupervised Domain Adaptation for Semantic Segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 1129-1139).

# Self-supervised Learning in UDA for Semantic Segmentation

## Self-Supervised Depth

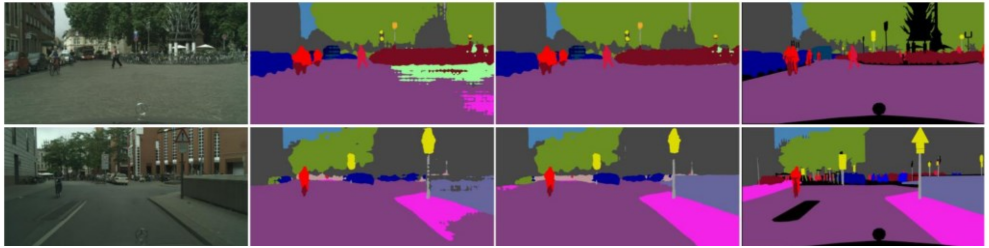


Data Augmentation Depth Based for Self-Training

# Self-supervised Learning in UDA for Semantic Segmentation

## Self-Supervised Depth

| Method | Road | Sidewalk | Building | Walls | Fence | Pole | T-light | T-sign | Vegetation | Terrain | Sky | Person | Rider | Car | Truck | Bus | Train | Motorbike | Bicycle | mIoU | Acc |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AdaptSegNet [49] | 86.5 | 36.0 | 79.9 | 23.4 | 23.3 | 23.9 | 35.2 | 14.8 | 83.4 | 33.3 | 75.6 | 58.5 | 27.6 | 73.6 | 32.5 | 35.4 | 3.9 | 30.1 | 28.1 | 42.4 | 85.6 |
| D4-AdaptSegNet + DBST | 93.1 | 53.0 | 85.1 | 42.8 | 27.3 | 35.8 | 43.9 | 18.5 | 85.9 | 39.0 | 89.9 | 63.0 | 31.6 | 86.6 | 39.8 | 36.7 | 0 | 42.4 | 35.0 | **50.0** | **90.3** |
| MaxSquare [5] | 88.1 | 27.7 | 80.8 | 28.7 | 19.8 | 24.9 | 34.0 | 17.8 | 83.6 | 34.7 | 76.0 | 58.6 | 28.6 | 84.1 | 37.8 | 43.1 | 7.2 | 32.2 | 34.5 | 44.3 | 86.9 |
| D4-MaxSquare + DBST | 92.9 | 51.2 | 84.7 | 43.5 | 22.2 | 35.7 | 42.5 | 20.0 | 86.2 | 42.0 | 90.0 | 63.7 | 33.0 | 86.9 | 45.5 | 50.9 | 0 | 42.2 | 41.4 | **51.3** | **90.3** |
| BDL [28] | 88.2 | 44.7 | 84.2 | 34.6 | 27.6 | 30.2 | 36.0 | 36.0 | 85.0 | 43.6 | 83.0 | 58.6 | 31.6 | 83.3 | 35.3 | 49.7 | 3.3 | 28.8 | 35.6 | 48.5 | 89.2 |
| D4-BDL + DBST | 93.2 | 52.6 | 86.4 | 44.1 | 31.2 | 36.5 | 42.4 | 36.1 | 86.3 | 41.0 | 89.8 | 63.3 | 37.4 | 86.3 | 42.8 | 57.8 | 0 | 40.3 | 37.9 | **52.9** | **90.7** |
| MRNET [69] | 90.5 | 35.0 | 84.6 | 34.3 | 24.0 | 36.8 | 44.1 | 42.7 | 84.5 | 33.6 | 82.5 | 63.1 | 34.4 | 85.8 | 32.9 | 38.2 | 2.0 | 27.1 | 41.8 | 48.3 | 88.3 |
| D4-MRNET + DBST | 93.2 | 51.6 | 86.1 | 45.9 | 24.5 | 37.9 | 47.4 | 40.4 | 85.3 | 37.5 | 89.6 | 64.7 | 39.8 | 85.8 | 41.1 | 53.2 | 8.9 | 17.1 | 33.4 | **51.7** | **90.0** |
| Stuff and things* [55] | 90.2 | 43.5 | 84.6 | 37.0 | 32.0 | 34.0 | 39.3 | 37.2 | 84.0 | 43.1 | 86.1 | 61.1 | 29.9 | 81.6 | 32.3 | 38.3 | 3.2 | 30.2 | 31.9 | 48.3 | 88.8 |
| D4-Stuff and things + DBST | 93.3 | 54.0 | 86.5 | 46.4 | 32.3 | 37.7 | 45.2 | 39.5 | 85.5 | 39.4 | 90.0 | 63.7 | 32.8 | 85.5 | 32.0 | 39.5 | 0 | 37.7 | 35.5 | **51.4** | **90.5** |
| FADA [54] | 92.5 | 47.5 | 85.1 | 37.6 | 32.8 | 33.4 | 33.8 | 18.4 | 85.3 | 37.7 | 83.5 | 63.2 | 39.7 | 87.5 | 32.9 | 47.8 | 1.6 | 34.9 | 39.5 | 49.2 | 88.9 |
| D4-FADA + DBST | 93.9 | 58.2 | 86.4 | 45.9 | 29.6 | 36.9 | 44.6 | 27.0 | 86.3 | 39.4 | 90.0 | 64.9 | 41.0 | 85.8 | 34.6 | 51.2 | 9.9 | 24.2 | 37.3 | **52.0** | **90.7** |
| LTIR [22] | 92.9 | 55.0 | 85.3 | 34.2 | 31.1 | 34.4 | 40.8 | 34.0 | 85.2 | 40.1 | 87.1 | 61.1 | 31.1 | 82.5 | 32.3 | 42.9 | 3 | 36.4 | 46.1 | 50.2 | 90.0 |
| D4-LTIR + DBST | 94.2 | 59.6 | 86.9 | 43.9 | 35.3 | 36.9 | 45.7 | 36.1 | 86.2 | 40.6 | 90.0 | 65.9 | 38.2 | 84.4 | 33.3 | 52.4 | 13.7 | 46.2 | 51.7 | **54.1** | **91.0** |
| ProDA [64] | 87.8 | 56.0 | 79.7 | 46.3 | 44.8 | 45.6 | 53.5 | 53.5 | 88.6 | 45.2 | 82.1 | 70.7 | 39.2 | 88.8 | 45.5 | 59.4 | 1.0 | 48.9 | 56.4 | 57.5 | 89.1 |
| D4-ProDA + DBST | 94.3 | 60.0 | 87.9 | 50.5 | 43.0 | 42.6 | 50.8 | 51.3 | 88.0 | 45.9 | 89.7 | 68.9 | 41.8 | 88.0 | 45.8 | 63.8 | 0 | 50.0 | 55.8 | **58.8** | **92.1** |



RGB UDA UDA + Depth GT

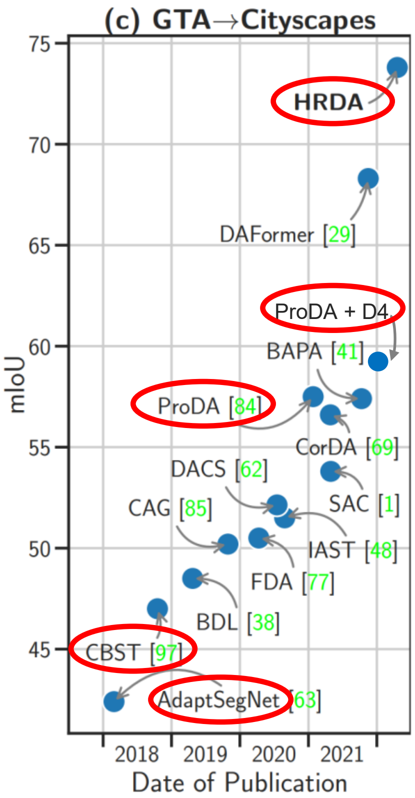
# Overview & Conclusion

# Overview of UDA Techniques

## Early 2022

- Self-ensembling
- Domain Alignment: Feature, Image or Output Level
- Pseudo-labelling and self-training
- Entropy minimization of target predictions
- Model distillation
- Self-supervised learning
- Co-training
- Curriculum learning
- Adversarial attacks

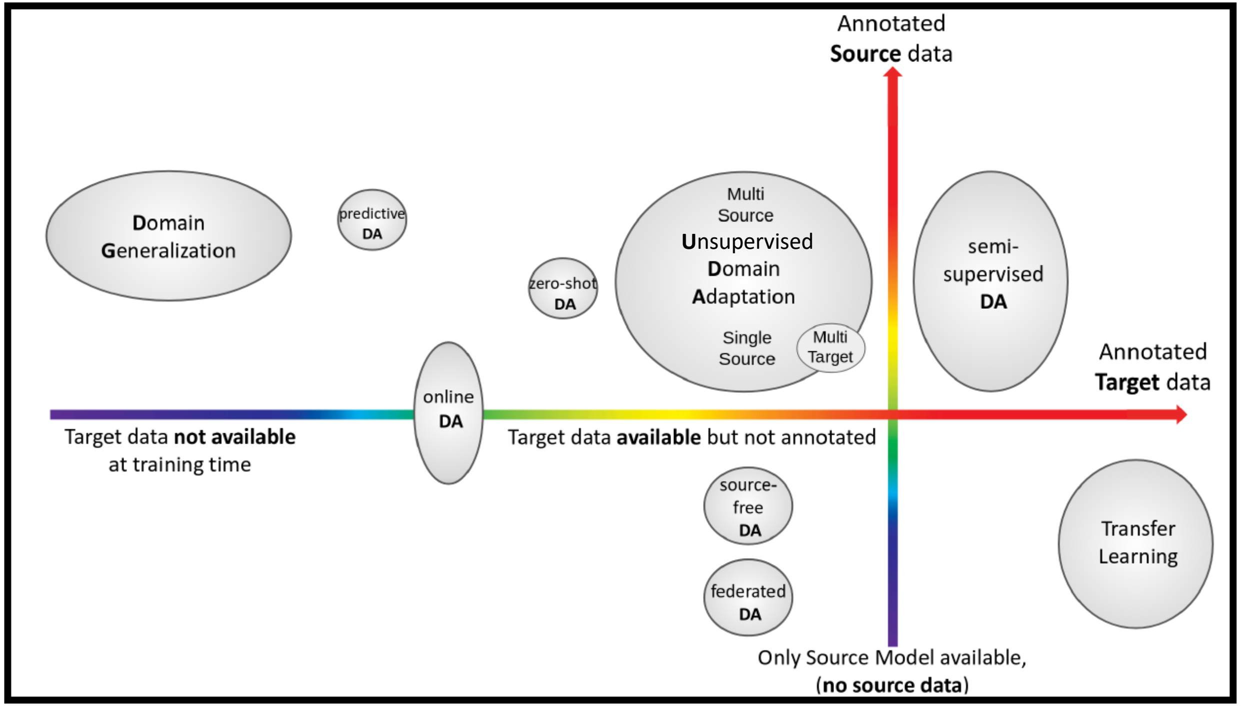
# Quantitative Results 2018-2022




| | Road | S.walk | Build. | Wall | Fence | Pole | Tr.Light | Sign | Veget. | Terrain | Sky | Person | Rider | Car | Truck | Bus | Train | M.bike | Bike | mIoU |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| GTA5 → Cityscapes | | | | | | | | | | | | | | | | | | | | |
| CBST [97] | 91.8 | 53.5 | 80.5 | 32.7 | 21.0 | 34.0 | 28.9 | 20.4 | 83.9 | 34.2 | 80.9 | 53.1 | 24.0 | 82.7 | 30.3 | 35.9 | 16.0 | 25.9 | 42.8 | 45.9 |
| DACS [62] | 89.9 | 39.7 | 87.9 | 30.7 | 39.5 | 38.5 | 46.4 | 52.8 | 88.0 | 44.0 | 88.8 | 67.2 | 35.8 | 84.5 | 45.7 | 50.2 | 0.0 | 27.3 | 34.0 | 52.1 |
| CorDA [69] | 94.7 | 63.1 | 87.6 | 30.7 | 40.6 | 40.2 | 47.8 | 51.6 | 87.6 | 47.0 | 89.7 | 66.7 | 35.9 | 90.2 | 48.9 | 57.5 | 0.0 | 39.8 | 56.0 | 56.6 |
| BAPA [41] | 94.4 | 61.0 | 88.0 | 26.8 | 39.9 | 38.3 | 46.1 | 55.3 | 87.8 | 46.1 | 89.4 | 68.8 | 40.0 | 90.2 | 60.4 | 59.0 | 0.0 | 45.1 | 54.2 | 57.4 |
| ProDA [84] | 87.8 | 56.0 | 79.7 | 46.3 | 44.8 | 45.6 | 53.5 | 53.5 | 88.6 | 45.2 | 82.1 | 70.7 | 39.2 | 88.8 | 45.5 | 59.4 | 1.0 | 48.9 | 56.4 | 57.5 |
| DAFormer [29] | 95.7 | 70.2 | 89.4 | 53.5 | 48.1 | 49.6 | 55.8 | 59.4 | 89.9 | 47.9 | 92.5 | 72.2 | 44.7 | 92.3 | 74.5 | 78.2 | 65.1 | 55.9 | 61.8 | 68.3 |
| HRDA | **96.4** | **74.4** | **91.0** | **61.6** | **51.5** | **57.1** | **63.9** | **69.3** | **91.3** | **48.4** | **94.2** | **79.0** | **52.9** | **93.9** | **84.1** | **85.7** | **75.9** | **63.9** | **67.5** | **73.8** |

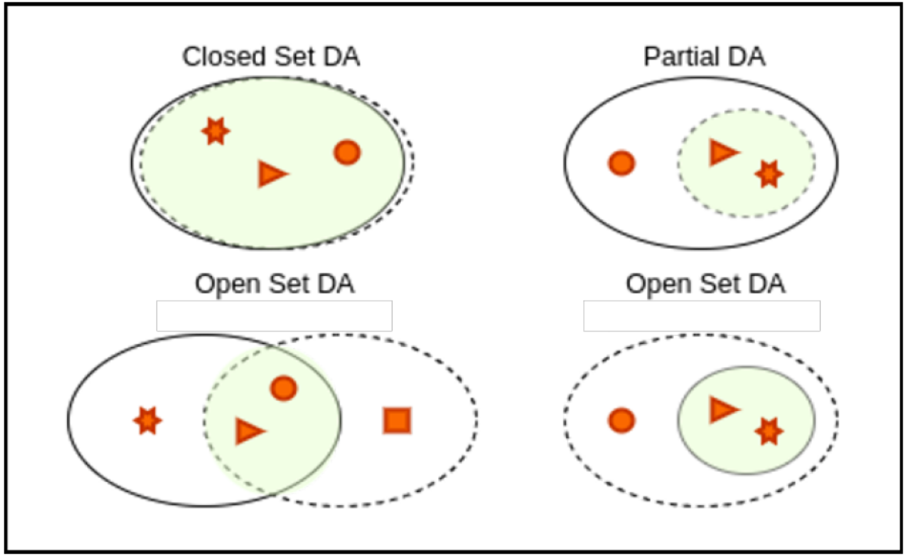
Hoyer, L., Dai, D., & Van Gool, L. (2022). HRDA: Context-Aware High-Resolution Domain-Adaptive Semantic Segmentation. *arXiv preprint arXiv:2204.13132*

# Overview of Adaptation Scenarios w.r.t. Labeled Data Availability




Csurka, G., Volpi, R., & Chidlovskii, B. (2021). Unsupervised Domain Adaptation for Semantic Image Segmentation: a Comprehensive Survey. *arXiv preprint arXiv:2112.03241*.

# Adaptation Scenarios
# w.r.t. Source and Target Label Sets Overlap




Source Domain

Target Domain

Csurka, G., Volpi, R., & Chidlovskii, B. (2021). Unsupervised Domain Adaptation for Semantic Image Segmentation: a Comprehensive Survey. *arXiv preprint arXiv:2112.03241*.