KIDELTA LEARNING
Scalable AI for Automated Driving

Final Event| March 10,2023

A Low-Complexity Approach for Domain Adaptation

Joshua Niemeijer, Jörg Schäfer | DLR

Effect of Domain Switch

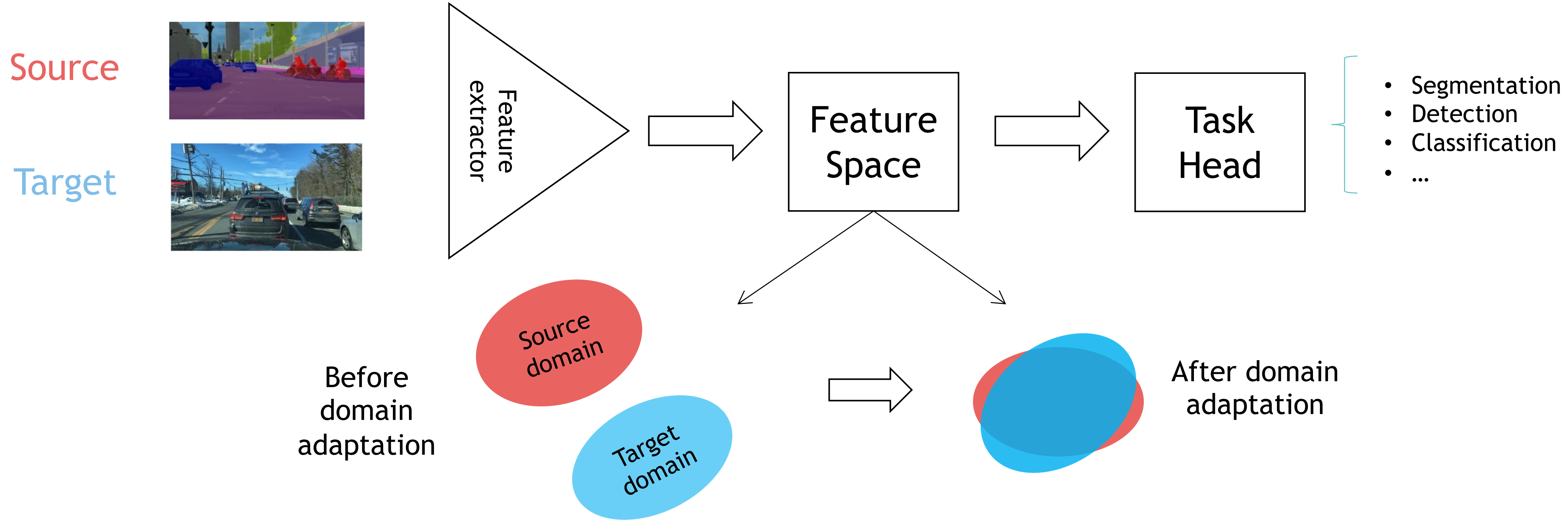
Source domain performance

Cityscapes

Target domain performance

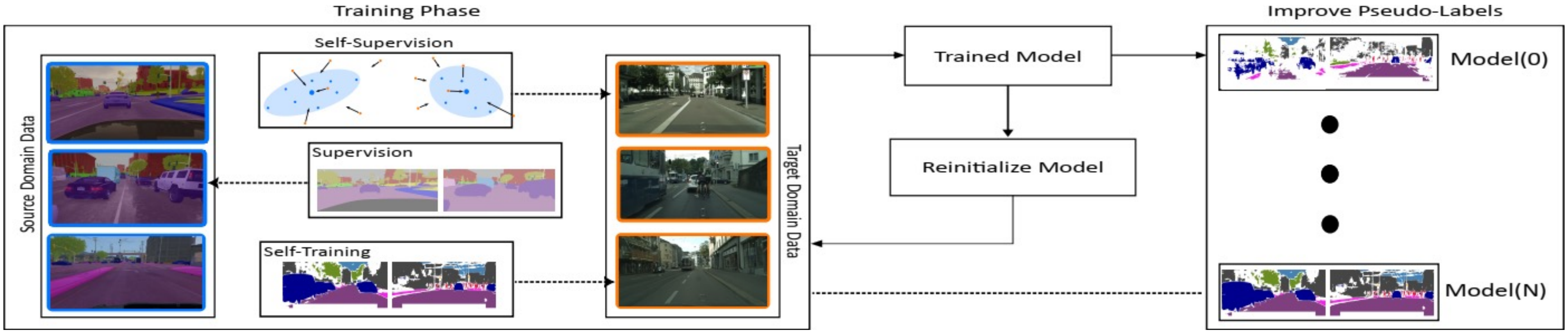
BDD

Domain Adaptation



- Source Domain: Data with annotation for training present
- Target Domain: No annotations present

A Low-Complexity Domain Adaptation Approach

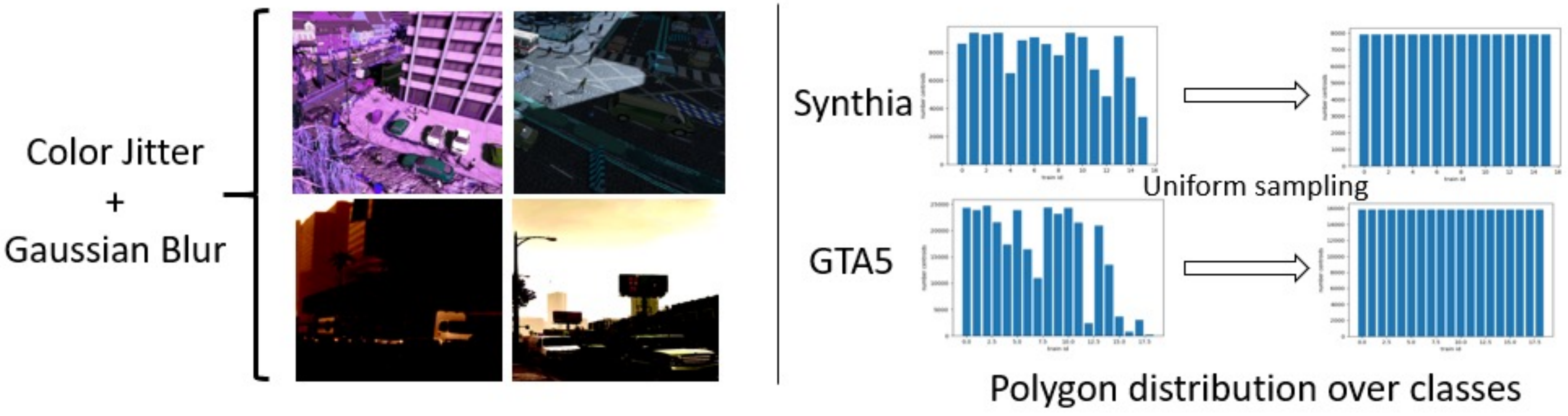


Generalizing Source Only Training

Semantic Self-Supervision

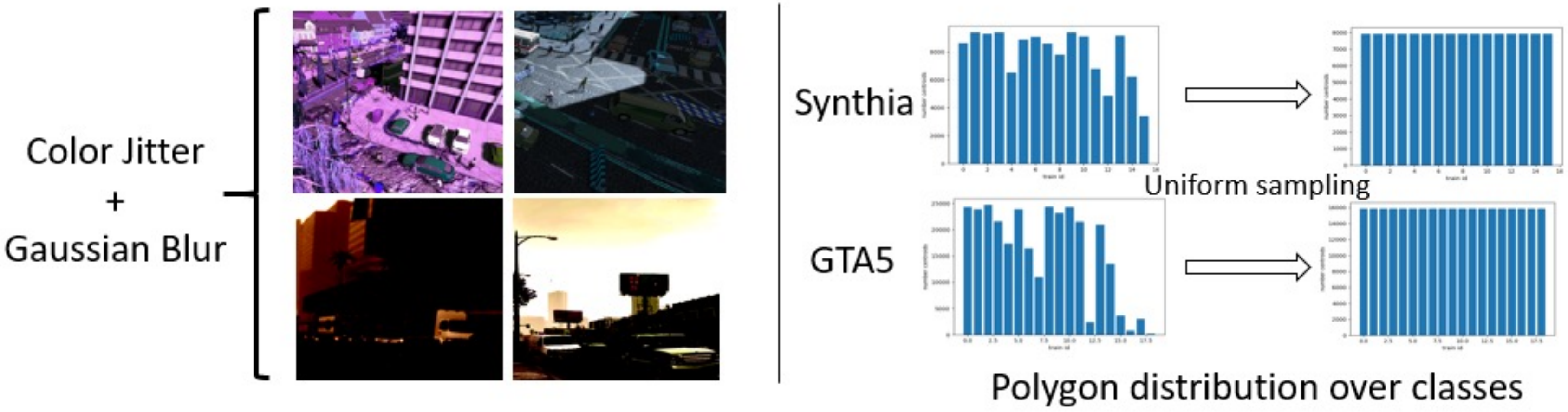
Self-Training

Generalizing Source Only Training



- Strong data augmentation
  - Cropping, Color Jitter and gaussian blurring
- Sampling: Zhu et. al.:“Improving semantic segmentation via video propagation and label relaxation” 2019
  - Crops are generated with an uniform class distribution
  - ➢ More weight to seldom classes but no overfitting due to strong augmentation
  - ➢ Less weight to often classes less overfitting on the simple synthetic data

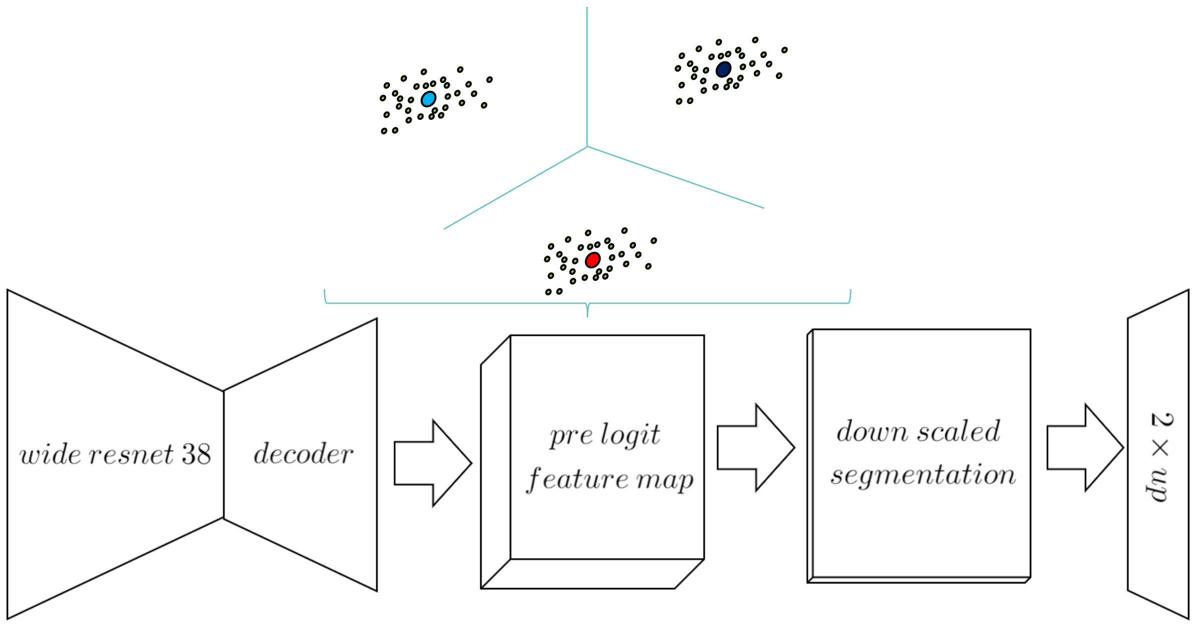
Generalizing Source Only Training



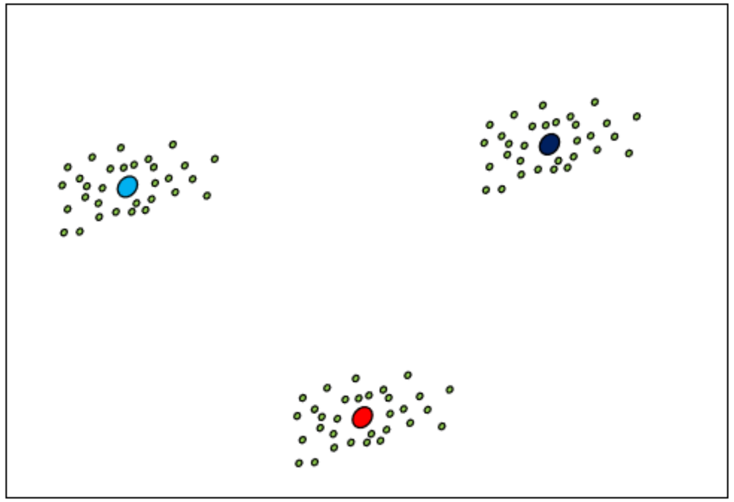
- When trained on GTA5 and tested on Cityscapes
  - With random cropping and horizontal flipping: 25.5% mIoU
  - With additional color Jitter and gaussian blurring: 38.8% mIoU
  - When additionally 50% of the epoch is sampled uniform: 41.4% mIoU
  - When additionally 100% of the epoch is sampled uniform: 44.5% mIoU

Semantic Self-Supervision

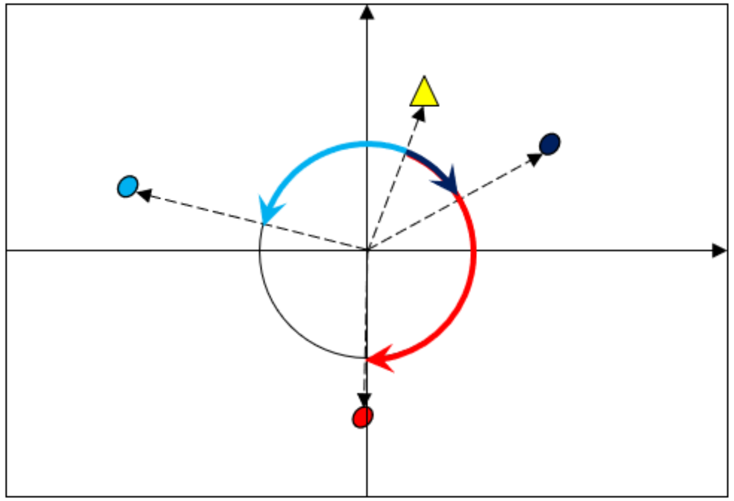
- Goal: Source and target domain class distribution alignment in pre-logit feature space
- Approach: Cluster pre-logit feature space to “class prototypes”
- However: For target domain the corresponding feature representation is not known
- Assumption: The closest class prototype is the correct one



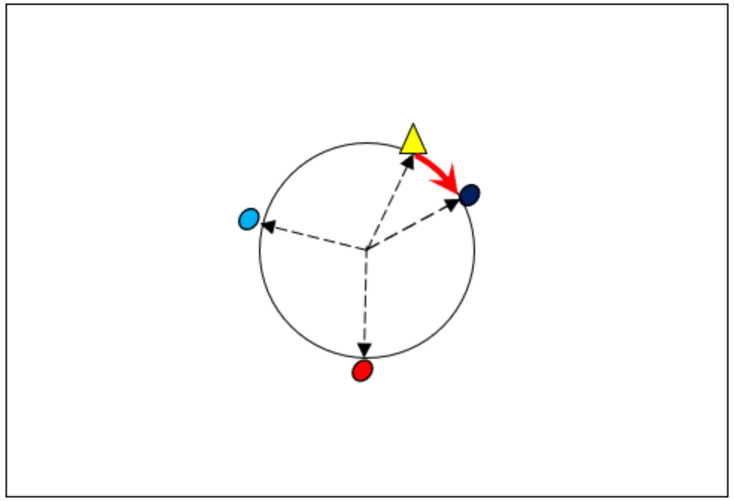
Semantic Self-Supervision



Determine class centroids
on source domain



Compute cosine similarity
between target representations
and class centroids



Minimize the entropy
in the similarity matrix

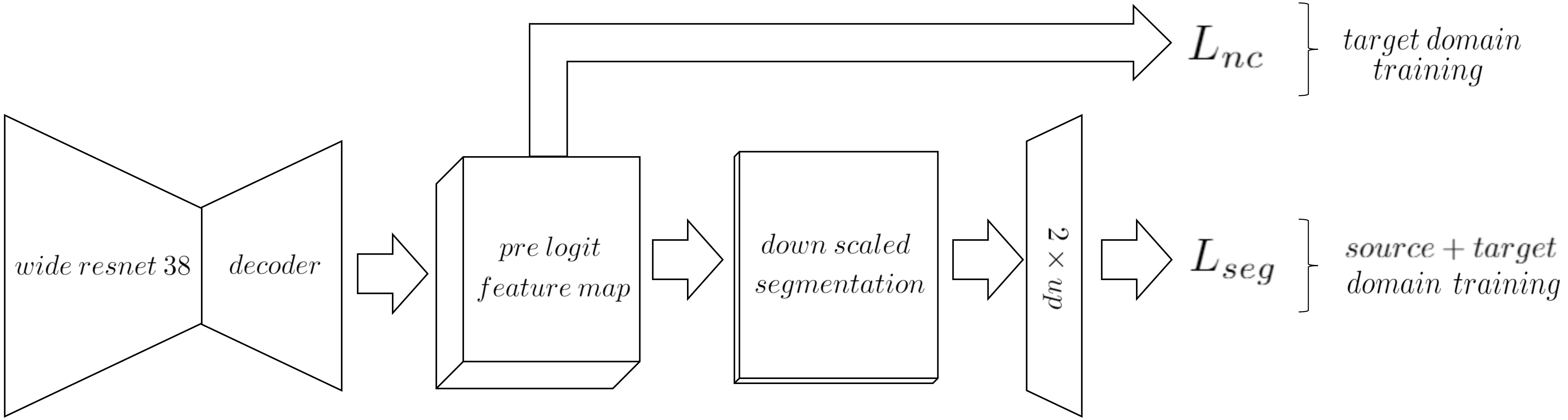
- Clustering is inspired by K. Saito et. al. “Universal Domain Adaptation through Self Supervision” 2020

Self-Training

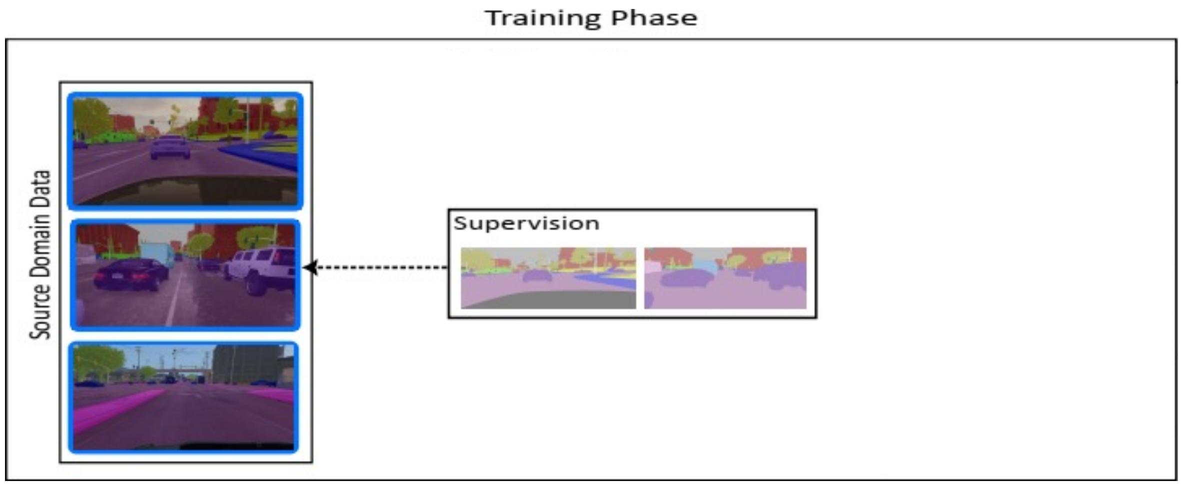
Threshold $\frac{\log(K)}{16}$

Domain Adaptation

- Half of the batch from labeled source domain and half from the target domain
- We apply our clustering loss here, as well
- ➢ Observation: The self-supervision improves the self-training

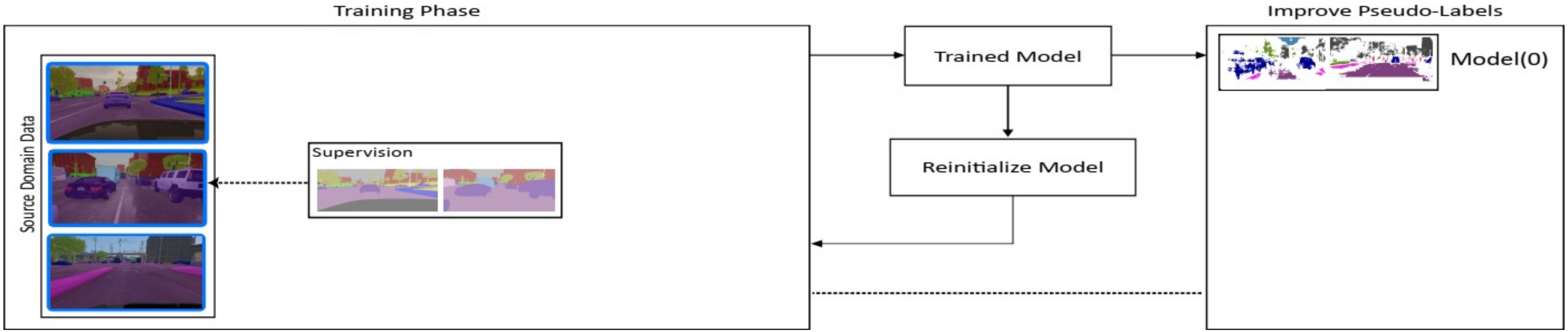


The Iterative Model



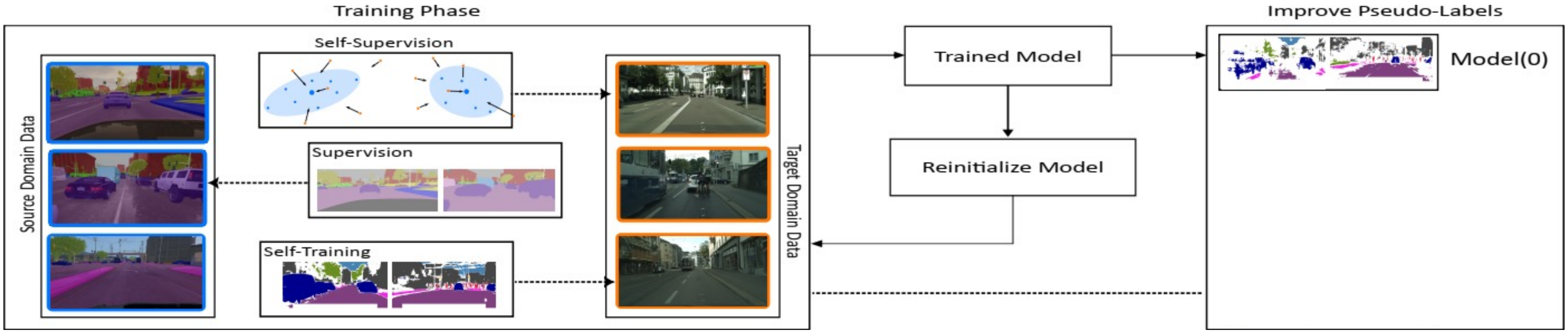
- Step1: Training on source domain yields first model

The Iterative Model



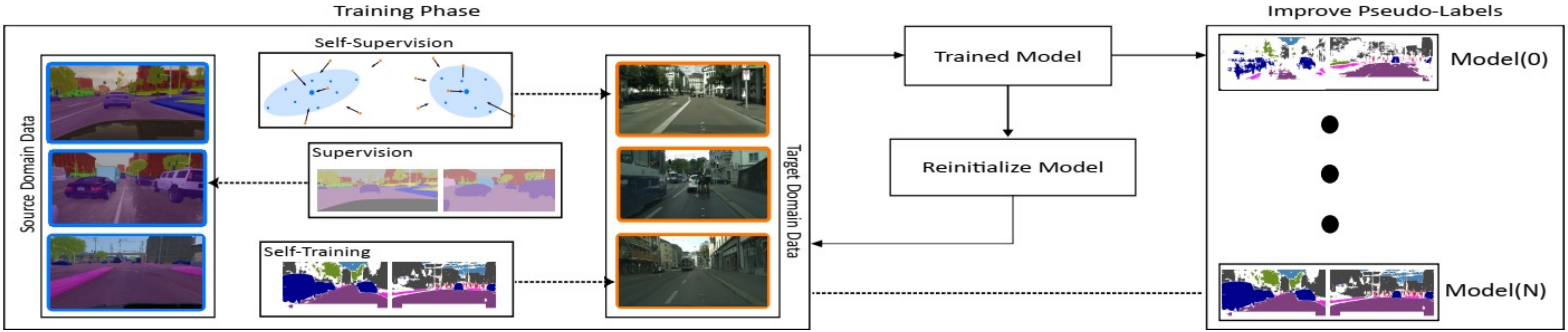
- Step1: Training on source domain yields first model
- Step2: Create pseudo labels with the model

The Iterative Model



- Step1: Training on source domain yields first model
- Step2: Create pseudo labels with the model
- Step3: Reinitialize the model and re-train on source and target domain
  - The self-training and the self-supervision are performed on the target domain

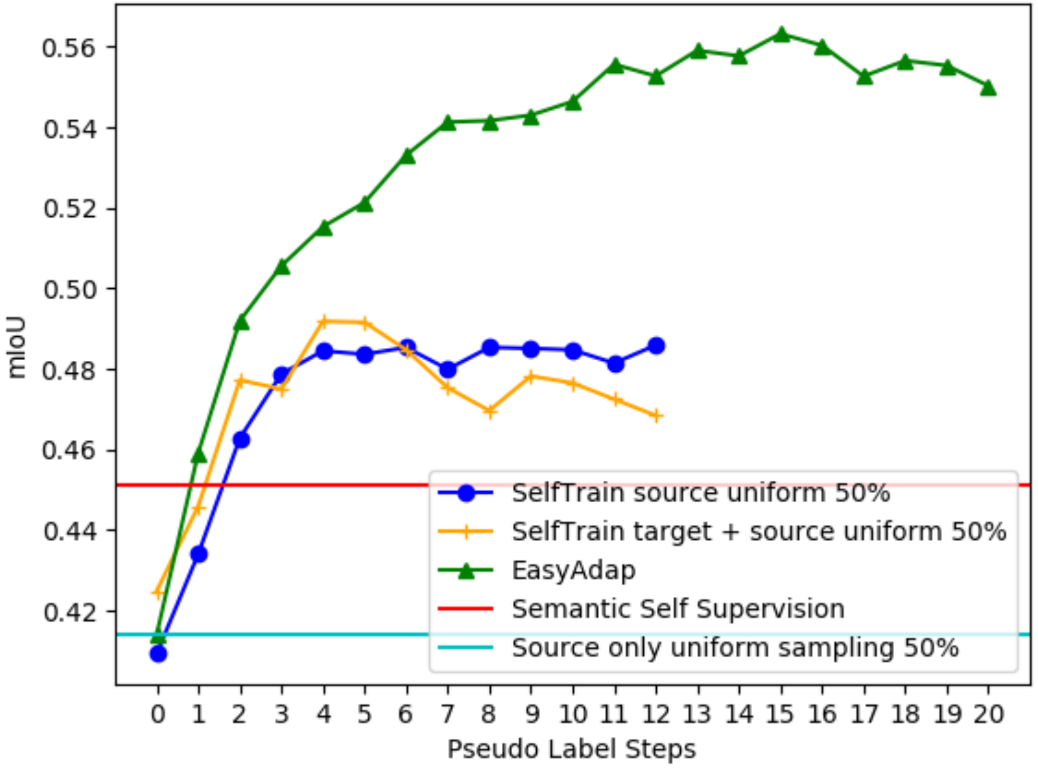
The Iterative Model



- Step1: Training on source domain yields first model
- Step2: Create pseudo labels with the model
- Step3: Reinitialize the model and re-train on source and target domain
  - The self-training and the self-supervision are performed on the target domain
- Step4: Repeat the process

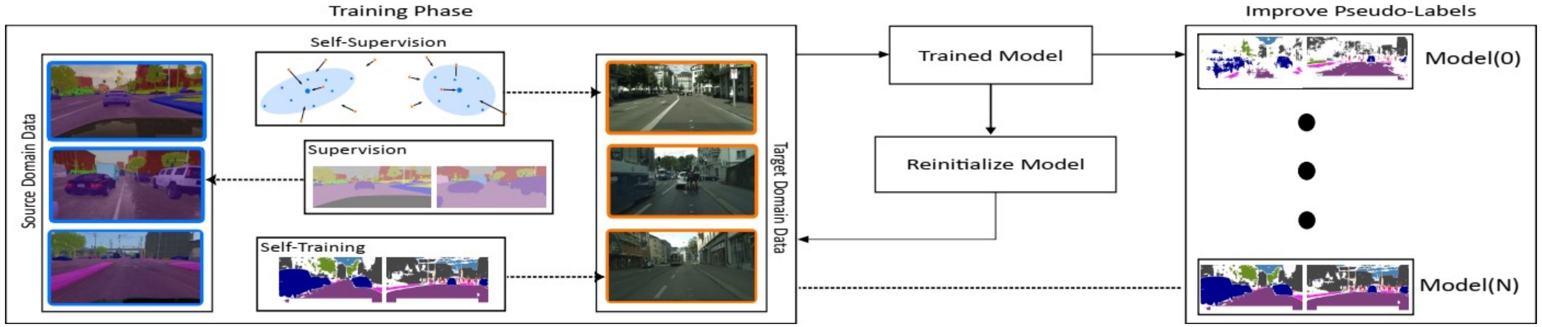
The Iterative Process (GTA5 to Cityscapes)

- Synergy between self-training and self-supervision
  - The self-training converges after 4 iterations
  - Additional semantic clustering: Improvement for 15 steps
  - Also the gradient is steeper
- Interpretation:
  - The self-training aligns the class distributions
  - Aligned class distributions lead to an improved clustering
  - The improved clustering achieves better pseudo labels
  - Better pseudo labels again lead to an improved clustering
  - ➢ Synergistic effect



Quantitative Evaluation: GTA5 to Cityscapes

- Our model is low in complexity compared to state of the art
- State of the art combines different loss functions and stages
- E.g. ProDA e.g. has got:
  - Three training stages
  - Combines: Self Training, Self Supervision, adversarial training …
  - ➢ High complexity comes with need for finetuning
- ➢ Our model is low in complexity



Approach	MIoU
Source DLv2	36.6%
AdapSeg	41.4%
CyCada	42.7%
CLAN	43.2%
APODA	45.9%
PatchAlign	46.5%
ADVENT	45.5%
BDL	48.5%
CBST	45.9%
MRKLD	47.1%
FADA	50.1%
CAG UDA	50.2%
SegUncertainty	50.3%
CLST	51.6%
SAC	53.8%
Coarse2Fine	56.1%
ProDA	57.5%
Source aug	**38.8%**
Self-Training	**49.2%**
Ours	**56.3%**

Qualitative Results

Before Domain Adaptation

After Domain Adaptation

- Published at „Conference on Robot Learning 2022“

KI DELTA LEARNING

Scalable AI for Automated Driving

Joshua Niemeijer | DLR | Joshua.Niemeijer@dlr.de

Jörg Schäfer | DLR | Joerg.Schaefer@dlr.de

Before Domain Adaptation

After Domain Adaptation

KI Delta Learning is part of the KI Familie and was developed by the VDA Leitinitiative Autonomous and Connected Driving.

KI FAMILIE

Gefördert durch:

Bundesministerium für Wirtschaft und Energie

aufgrund eines Beschlusses des Deutschen Bundestages

www.ki-deltalearning.de @KI_Familie KI Familie