# Domain Adaptation on Road Scene Segmentation under Different Weather Conditions

Tz-Ying Wu Shang-Wei Hung Chi-Hsin Huang Shiang Hu Li-Kuo Lin
Hsuan Chang
Department of Electrical and Computer Engineering
University of California San Diego
{tzw001, shung, chh286, shhu, lil001, hsc075}@ucsd.edu

## Abstract

*Reasoning the semantic meaning of road scene is essential for UAV systems to plan how to act properly. However, the visual observation of road scene differs a lot under different weather conditions, which may cause self-driving system drastically fail. In this project, we aim at learning unsupervised domain adaptation on road scene segmentation under different weather conditions, which is less studied in literature. We train semantic segmentation on a certain weather, and adopt unsupervised adversarial training to transfer the segmentation model to the target weather condition. First, we make use of synthetic dash-cam data from SYNTHIA datasets, to explore the domain shift of cross weather adaptation. Next, in order to examine our method could be applied to the real image, we conduct experiment in the MVD dataset. Finally, we perform an experiment using datasets from CARLA simulator for autonomous driving systems to prove that our method could be generally applied to different datasets for future applications. We show by experiment that our proposed method effectively align cross-weather data in feature space, and successfully adapt segmentation model to different weather conditions.*

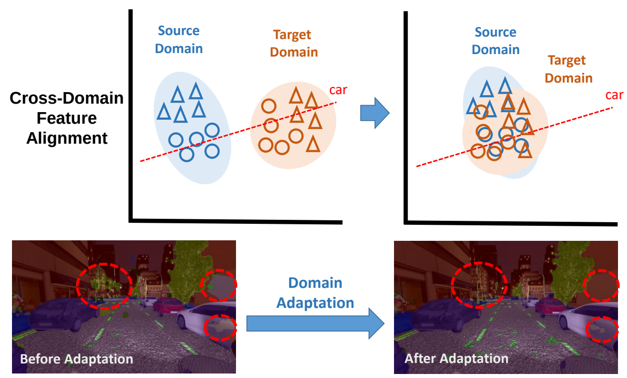


Figure 1. Overview of domain adaptation for segmentation of road scene.

## 1. Introduction

Recent developments of technologies in computer vision, deep learning, and articial intelligence have led to the raise of building autonomous driving systems and robotic navigation to mapping and categorizing the natural world. From recognizing particular objects to understand the corresponding driving environments, segmentation of road scene is among the key module for a successful autonomous driving system. With a sufficient amount of pixel-wise annotations, current computer vision algorithms already show promising performances on the above task. However, collecting and labeling training data for pixel-wise semantic segmentation task costs a lot of human labor to annotate these data, which is a time-consuming and expensive process. In addition, when one applies pre-trained segmenters to a road scene in different cities, driving routes and weathers, which are different from training data, the performance of results would be degraded due to dataset biases. The reason is that current approaches are sensitive to the large variations due to changes in appearances, objects and lighting. Thus, how to suppress the dataset bias would be important when there is a necessity to apply road scene segmenters to road scenes in different conditions.

To alleviate this problem, people propose an unsupervised learning framework for transferring semantic segmentation across image domains, making it particularly appealing to learn to share and transfer information between related settings. Recently, Chen et al,[3] presented an unsupervised domain adaptation method for road scene semantic segmentation across different cities. Chen et al,[4] presented a model Reality Oriented ADaptation Network (ROAD-Net) for semantic segmentation of urban scenes by learning from synthetic data. Hoffman et al.[5] propose fully convolutional networks with domain adversarial training on domain shifts between different cities, seasons and from synthetic to real. These works focuses on road scene segmentation with different road scenes, city environment and seasons; however, cross-weather adaptation is less studied. Thus, how to suppress the dataset bias would be critical when there is a need to deploy road scene segmenters to different weathers.

In this work, we propose unsupervised transfer learning for a road scene segmentation under different weather conditions. Here, we focus on global domain adaptation to align cross domain data in feature space so that we can apply the same segmentation model. Our proposed model is able to adapt a pre-trained segmentation model from source weather to target weather, while only the collection of unlabeled road scene images of one kind of the weather. We conduct experiments to demonstrate the issue of different weather discrimination using a state-of-the-art semantic segmenter. Then, we will verify the effectiveness of our proposed method across different datasets domain adaptation task under different weathers. By comparing it with baseline and upper bound, our proposed method achieves apparently better performance than the baseline in most cases.

We evaluate our approach using multiple large-scale datasets. First, we make use of synthetic dash-cam data from SYNTHIA datasets, to explore the domain shift of cross weather adaptation. Next, we perform an experiment using datasets from CARLA simulator for autonomous driving systems to prove that our method could be generally applied to different datasets for future applications. Finally, in order to examine our method could be applied to the real image, we conduct experiment in the MVD dataset to demonstrate that our method has potential to apply in the real autonomous driving systems and robotic navigation systems.

The main contributions of our project are summarized as follow:

- We use experimental result to point out the problem that segmentation performance degrades a lot for different weather conditions, which is less studied in previous literature.

- There is few existing dataset providing weather annotations. We collect a road scene segmentation dataset by CARLA simulator, which contains 6k images under 6 weather conditions.

- We show by experiment that unsupervised adversarial training effectively align cross-weather data in feature space, and successfully adapt segmentation model to different weather conditions.

# 2. Related Works

## 2.1. Semantic Segmentation

Semantic segmentation is a highly active field and leads the recent breakthrough in computer vision with large amount of methods proposed. Traditional works in semantic segmentation are typically based on manually designed image features. With the recent development of deep learning, learned representation demonstrates its power in many computer vision tasks. Here, we briefly review some of the works which focus on Convolutional Neural Networks (CNN)-based semantic segmentation, which has been successfully applied to predict dense pixel-wise semantic labels.

Long et al.[7] formulates semantic segmentation as a per-pixel classication problem by building fully convolutional networks that take input of arbitrary size and produce correspondingly-sized output with efficient learning and inference. Badrinarayanan et al.[1] present a novel and practical deep fully convolutional neural network architecture for semantic pixel-wise segmentation termed SegNet. The main motivation behind SegNet was the need to design an efficient architecture for road and indoor scene understanding which is efficient both in terms of memory and computational time. Ronneberger etal.[9] present a network and training strategy that relies on the strong use of data augmentation to use the available annotated samples more efficiently in biomedical segmentation applications. DeepLab [2] system re-purposes networks trained on image classification to the task of semantic segmentation by incorporating Conditional random led (CRF) with CNN to reason about spatial relationship. Zhao etal. [12] used Pyramid Pooling Module with the proposed pyramid scene parsing network (PSPNet) to encode the global and local context, which achieved state-of-the-arts results on multiple datasets.

Most of models have good performance in a supervised setting, but performance can be surprisingly poor under domain shifts that appear mild to a human observer. For example, training on different cities, geographic regions and weather conditions may result in significantly degraded performance due to pixel-level distribution shift. Hoffman et al. [5] presented fully convolutional networks with domain adversarial training for global domain alignment, while leveraging class-aware constrained multiple instance loss for transferring spatial layout. They demonstrated the effectiveness of their method on domain shifts between different cities, seasons and from synthetic to real. However, cross-weather adaptation is less studied. In this work, we aim at learning domain adaptation on road scene segmentation under different weather conditions.

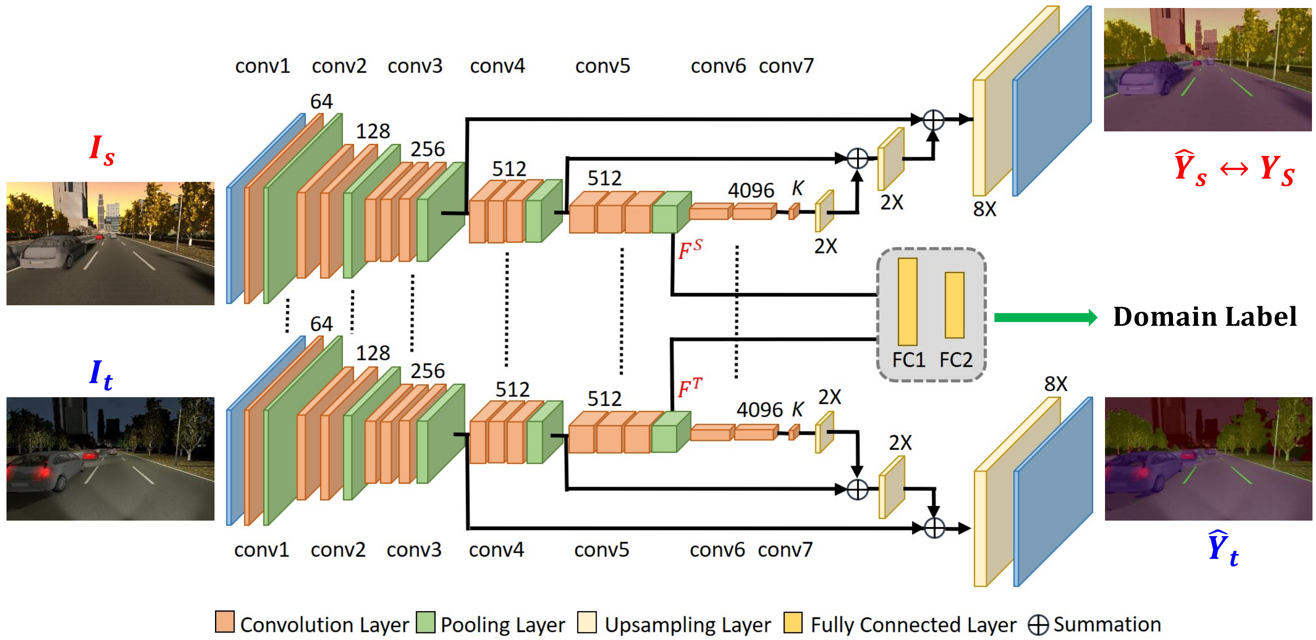


Figure 2. Our model is composed of two components, segmentation model and domain classifier. We adopt fcn8s as our segmentation model and a two-layer neural network as the domain classifier. For the segmentation model part, it is a two-stream model: one for source domain image $I_S$ and one for the target domain image $I_T$ which share the same weights. The output of the segmentation model for two domains are the semantic segmentation prediction $\hat{Y}_S$ and $\hat{Y}_T$ respectively. Its objective is to segment the image and also learn the feature that domain classifier cannot discriminate. On the other hand, the objective of domain classifier is to discriminate features from both domains, which is designed as a binary classifier. The input of the domain classifier is conv5 feature from the segmentation model, and the output of which is the prediction of the domain label.

### 2.2. Domain Adaptation

In conventional machine learning, people train and test data which are sampled independently from an identical distribution. However, in real world scenarios, this strategy often leads to a significant performance drop on the test data when applying the trained model. Thus, domain adaptation aims to decrease the impact of distribution mismatch such that the generalization ability of the learned model can be improved on the target domain. In computer vision, domain adaptation has been widely investigated as an image classification problem. Recently, the community start to pay attention to domain shift problem in semantic segmentation. Here, we focus on domain adaptation on road scene segmentation.

Hoffman et al. [5] presented fully convolutional networks with domain adversarial training for global domain alignment to have good performance different settings on multiple large-scale datasets, including adapting across various real city environments, different synthetic sub-domains, from simulated to real environments. Chen et al, [3] presented an unsupervised domain adaptation method for road scene semantic segmentation, which alleviates cross-domain discrimination on road scene images across different cities. Based on Generative Adversarial Network, Huang et al, [6] proposed an image-to-image translation network for generating large-scale trainable data for vehicle detection algorithms. Chen et al, [4] presented a new model Reality Oriented ADaptation Network (ROAD-Net) for semantic segmentation of urban scenes by learning from synthetic data.

These works focuses on road scene segmentation with different city environment; however, cross-weather adaptation is less studied. In this work, we aim at learning domain adaptation on road scene segmentation under different weather conditions.

## 3. Our Method

In this section, we first formulate the problem of domain adaptation on semantic segmentation. Next, we describe the details of the segmentation model and domain adaptation respectively. Finally, we provide implementation details for reproduction.

### 3.1. Problem Formulation

In this project, we aim at learning unsupervised domain adaptation on semantic segmentation under different weather conditions. For this domain adaptation problem, we first need to learn our main task, semantic segmentation,

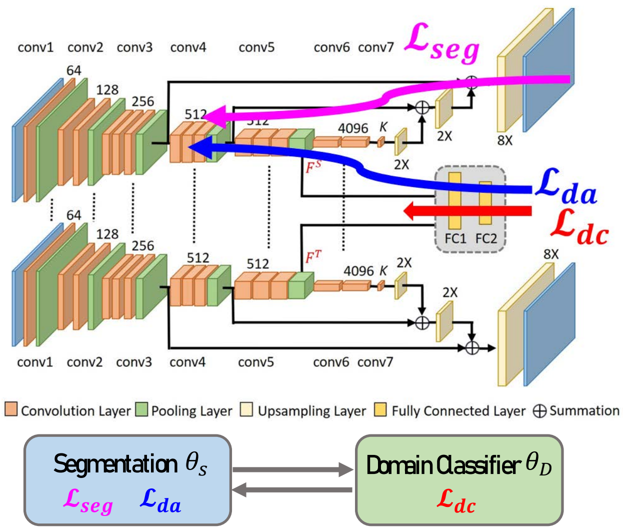


Figure 3. Our model consists of two parts: segmentation model and domain classifier. The objective of the former is to segment the image and also learn the feature that domain classifier cannot discriminate, so it is optimized with $\mathcal{L}_{seg}$ (pink) and $\mathcal{L}_{da}$ (blue). On the other hand, the objective of domain classifier is to discriminate features from both domains, which is designed as a binary classifier. Thus, it's optimized with $\mathcal{L}_{dc}$ (red). These two components have adversarial objectives, and they are trained iteratively.

on a certain weather (source domain $S$), and then we consider leveraging the learnt knowledge to the target weather condition (target domain $T$). Semantic segmentation in source domain aims at predicting semantic labels for each pixel $\hat{Y}_S \in \mathbb{R}^{H \times W}$ in the given input image $I_S \in \mathbb{R}^{H \times W}$, which is optimized with the ground truth semantic labels of the image $Y_S \in \mathbb{R}^{H \times W}$. Note that $H$ and $W$ denote image height and width respectively. Likewise, for target domain image $I_T \in \mathbb{R}^{H \times W}$, our goal is to predict the semantic labels for each pixel $\hat{Y}_T \in \mathbb{R}^{H \times W}$ in the given image. However, we can't access to the ground truth label $Y_T \in \mathbb{R}^{H x W}$ during training since we consider transferring segmentation model in an unsupervised fashion.

### 3.2. Semantic Segmentation

We consider Fully Convolutional Network (FCN) proposed by [7] as our segmentation model. To be more specific, we adopt FCN-8s model which is based on VGG-Net [11] with 16 layers. VGG-Net is a commonly adopted Convolutional Neural Network (CNN), which extracts coarse-to-fine features of the input image layer by layer. The output layer of FCN-8s is formed of the ensemble of the output of Pool3, Pool4 and Conv7 layers, which capture different hierachies of features of the input image. Since the feature map is downsized twice after each convolutional layer, we perform upsampling to resize feature maps to the original image size correspondingly. We sums the 2x upsampled conv7 (convolutionalized fc7) with pool4, upsamples them with a stride 2 transposed convolution and sums them with pool3, and applies a transposed convolution layer with stride 8 on the resulting feature maps followed by a Softmax layer to obtain the segmentation map $R \in \mathbb{R}^{K \times H \times W}$, where $K$, $H$ and $W$ denotes number of semantic classes, image height and image width respectively. For each pixel in the image, the semantic label prediction is obtained by selecting the class which has largest probability as follows,

$$\hat{y}_{hw} = \underset{k \in |K|}{\arg\max}\, p_{\theta_S}(R_{hw})_k. \tag{1}$$

The segmentation model is parameterized with $\theta_S$. and is optimized with pixel-wise cross-entropy loss, as follows,

$$\mathcal{L}_{seg} = -\sum_{I_s \in S} \sum_{h \in H} \sum_{w \in W} \sum_{k \in K} Y^k_{h,w} log(p_{\theta_S}(R^S_{hw})) \tag{2}$$

### 3.3. Domain Adaptation

Our goal is to learn a semantic segmentation model which can be adapted to the unlabeled target domain $T$. If there is no domain shift between the source and target domains then one could simply apply the source model directly to the target without an adaptive approach. However, there is commonly a difference between the distribution of the source labeled domain and the target test domain. In order to minimize the domain shift between representations of the source and target data, adversarial learning is adopted, whereby simultaneously a domain classifier is trained to best distinguish the source and target distributions and the representation space is updated according to the inverse objective.

Figure 2 shows the overall framework of our model, which is composed of two components, segmentation model and domain classifier. The segmentation model is a two-stream model: one for source domain and one for the target domain which share the same weights. Its objective is to segment the image and also learn the feature that domain classifier cannot discriminate. On the other hand, the objective of domain classifier here is to discriminate features from both domains, which is designed as a binary classifier. The objective of the two components is adversarial, and they are trained iteratively as shown in Figure 3. We describe the details of these components below.

**Domain Classifier.** The domain classifier is designed as a two-layer neural network, which is parameterized with $\theta_D$. Its objective is to distinguish features from different domains; therefore, the domain classification loss is defined as a binary cross-entropy loss, as follows,

$$\begin{aligned}\mathcal{L}_{dc} = &-\sum_{I_S \in S} \sum_{h \in H} \sum_{w \in W} log(p_{\theta_D}(F^S_{hw})) \\ &-\sum_{I_\mathcal{T} \in \mathcal{T}} \sum_{h \in H} \sum_{w \in W} log(1 - p_{\theta_D}(F^\mathcal{T}_{hw})),\end{aligned} \tag{3}$$

where $F^S_{hw}$ and $F_{hw}$ denote the source and target representation of each units from Conv5 layer, respectively.

**Segmentation Model.** As we introduce in Sec.3.2, FCN-8s based on VGG-16 is adopted as our segmentation model. The objective of the segmentation model is to segment the image and also learn the feature that domain classifier cannot discriminate.

As we define the domain classification loss above, we define the inverse domain classification loss, $\mathcal{L}^{inv}_{dc}$ as follows,

$$\begin{aligned}\mathcal{L}^{inv}_{dc} = &-\sum_{I_S \in S}\sum_{h\in H}\sum_{w\in W} log(1 - p_{\theta_D}(F^S_{hw})) \\ &-\sum_{I_\mathcal{T} \in \mathcal{T}}\sum_{h\in H}\sum_{w\in W} log(p_{\theta_D}(F^\mathcal{T}_{hw}))\end{aligned} \tag{4}$$

If we only consider $\mathcal{L}^{inv}_{dc}$ for domain adversarial training, the training procedure may be unstable. Hence, we consider balancing the both losses as our domain adversarial loss as follows,

$$\mathcal{L}_{da} = \mathcal{L}_{dc} + \mathcal{L}^{inv}_{dc}. \tag{5}$$

With the above loss terms defined, the overall loss of the segmentation model can be written as,

$$\mathcal{L}_{total} = \mathcal{L}_{seg} + \lambda \mathcal{L}_{da} \tag{6}$$

where $\lambda$ is the regularization term for the global domain adversarial loss, which is 0.00001 in our case.

**Iterative Training.** As shown in Figure 3, with these definitions, we may now describe the alternating minimization procedure.

$$\min_{\theta_D} \ \mathcal{L}_{dc} \tag{7}$$

$$\min_{\theta_S} \ [\mathcal{L}_{seg} + \lambda \mathcal{L}_{da}] \tag{8}$$

Optimizing these two objectives iteratively amounts to learning the best possible domain classifier for relevant image regions (Eq. 7) and then using the loss of that domain classifier to inform the training of the image representations so as to minimize the distance between the source and target domains (Eq. 8).

### 3.4. Implementation Details

We use FCN-8s as our based segmentation model. The input size of FCN-8s is 380x640 for the SYNTHIA dataset, while for the CARLA/MVD dataset, the input size is 600x800. For the domain classifier, the first fully connected layer has size 512 the size of the second fully connected layer is 64 for the SYNTHIA/CARLA dataset and 128 for the MVD dataset respectively. Our best model is chosen according to the evaluation on the validation set. We train our segmentation model with learning rate 0.00024.

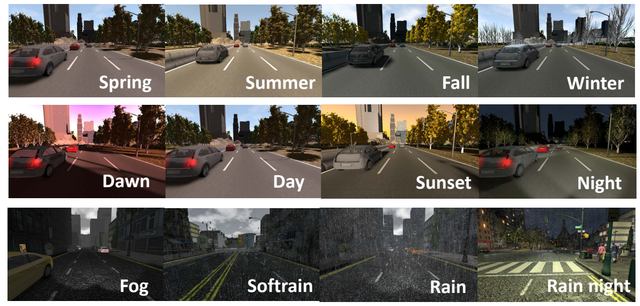


Figure 4. SYNTHIA dataset.

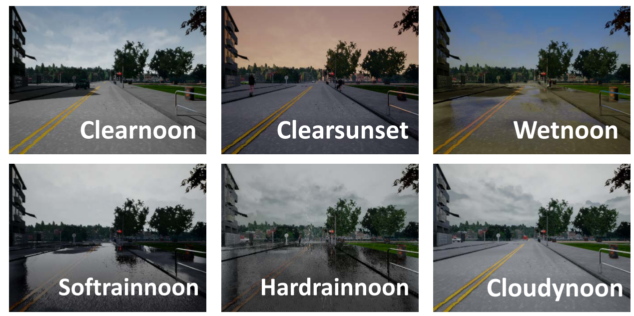


Figure 5. CARLA dataset.


Figure 6. MVD dataset.

## 4. Dataset

We briefly introduce the datasets we use in our experiment. Existing road scene datasets like Synthia [10], MVD [8] provide labels for weather conditions. We use these datasets for our experiment. Furthermore, We build up a synthetic dataset with CARLA [1] simulators.

**SYNTHIA** is a synthetic collection of images of road scenarios, containing 13 classes with different scenarios and sub-conditions. There are 7 sequences, covering different scenarios (European style town, modern city, highway and green areas) with several sub-sequences, such as seasons (Spring, Summer, Fall, Winter), weathers condition (sunny, cloudy, rain, snow, fog), and illuminations condition (Sunset, Dawn, Night). Also, there are a variety of dynamic objects, including cars, pedestrians and cyclists. These frames are captured by 8 RGB cameras forming a binocular 360 camera, 8 depth sensors. In this task, we take SEQS-05

[1]http://carla.org/

(NewYork-like city), providing 8k images random images from all the sequences, as source domain data.

**CARLA** is a simulator to support development, training, and validation of autonomous driving systems. The simulation platform provides open digital assets (urban layouts, vehicles, buildings,) and supports flexible specification of sensor suites, control of all dynamic and static actors, maps generation and different environmental conditions, including weather. In this project, we build up our own synthetic dataset with CARLA, as source domain data.

**MVD** is a diverse street-level imagery dataset with pixelaccurate for understanding street scenes around the world. There are 25k high-resolution real images with 100 instance-specifically annotated categories and 152 object categories. Also, there are variety of weather, season, time of day, camera, and viewpoint. However, weather annotation is not provided; therefore, we label the weather conditions, sunny and cloudy, by ourselves.

## 5. Experiment

### 5.1. Evaluation Metric

For each experiment, we use mean intersection-over-union (mIoU) as metrics to evaluate performance. Let $n_{ij}$ be the number of pixels of class $i$ predicted as class $j$, let $t_j = \sum_j n_{ij}$ be the total number of pixels of class $i$, and let $N$ be the number of classes. The definition is shown as follows:

$$\text{mIoU} = \frac{1}{N} \cdot \frac{\sum_i n_{ii}}{t_i + \sum_j n_{ij} - n_{ii}} \tag{9}$$

### 5.2. Setting Variants

**Supervised.** We use a fully-supervised learning to establish a strong baseline as the upper bound of adaptation experiment.We train images with fine annotations on source weather as training set. And, we test our model to the same weather to evaluate performance as the upper bound performance.

**Before Adapt.** We apply the source domain model which is pre-trained on source weather on target weather without any adaptation as baseline. The different visual appearances across weathers would dramatically impact the accuracy of the segmenter.

**After Adapt.** We adapt the domain adversarial learning method to adapt source domain model in an unsupervised learning. Then, we use this model with adaptation to train the different target weather to evaluate the transfer learning performance.

### 5.3. Synthetic Experiment

We conduct experiments to demonstrate the issue of weather discrimination even using a state-of-the-art semantic segmenter on synthetic dataset. Then, we will verify the

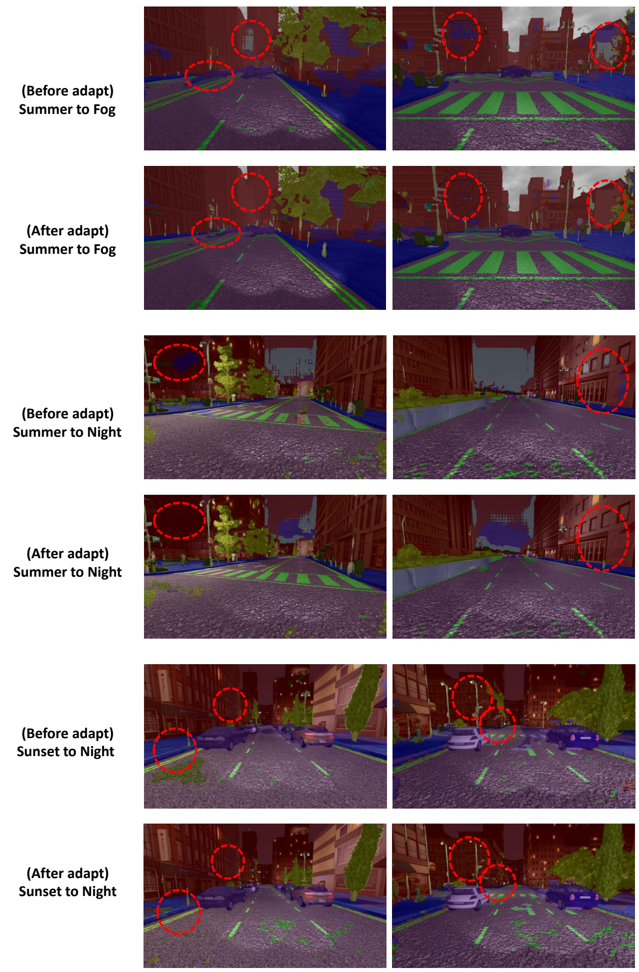


Figure 7. Visualization of SYNTHIA experiment.

effectiveness of our proposed unsupervised learning method test on different weathers datasets domain adaptation task. By comparing it with a supervised baseline, we show that that our method would achieve comparable performances as the supervised training methods in most cases. Then, we perform an experiment using datasets from CARLA simulator for autonomous driving systems to prove that our method could be generally applied to different datasets for future applications.

**Cross Weathers on SYNTHIA Dataset** In first experiment, we analyze adaptation across weather patterns by using SYNTHIA dataset which has synthetic images available along with weather annotations. We first perform supervised training by adapting the same weather both in source and target domain with fully annotated training data. Then, we also conduct experiments for before adaptation and after adaptation case for training on one season and evaluating on another unannotated season, including summer, rain, night, fog and sunset. We report quantitative comparisons of performance supervising learning, before and after adap-

Table 1. **Adaptation across weathers on SYNTHIA.** We study the cross weather performance using SYNTHIA dataset. We report quantitative of supervising learning and comparisons of performance before and after adaptation for training on one weather and evaluating on another unannotated weather.

| Method | Source | Target | Sky | Building | Road | Sidewalk | Fence | Vegetation | Pole | Car | Traffic sign | Pedestrian | Lane marking | Traffic light | mIoU |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Supervised | Summer | Summer | 98.23 | 94.79 | 97.94 | 93.28 | 96.70 | 82.75 | 55.53 | 93.87 | 51.30 | 85.70 | 87.16 | 53.30 | 82.54 |
| Before Adapt | Summer | Rain | 95.41 | 51.28 | 62.98 | 19.21 | 19.32 | 11.06 | 6.25 | 30.23 | 7.31 | 7.02 | 36.30 | 11.11 | **25.54** |
| After Adapt | Summer | Rain | 95.06 | 42.59 | 59.71 | 20.38 | 26.89 | 23.28 | 4.49 | 29.96 | 9.24 | 8.62 | 21.51 | 11.57 | **27.18** |
| Supervised | Rain | Rain | 97.62 | 93.57 | 97.53 | 89.58 | 95.49 | 78.27 | 36.34 | 92.82 | 43.66 | 83.04 | 83.96 | 56.15 | 79.00 |
| Before Adapt | Summer | Fog | 84.82 | 74.44 | 80.48 | 67.83 | 33.77 | 32.77 | 29.83 | 31.53 | 28.38 | 21.38 | 60.51 | 32.32 | **41.30** |
| After Adapt | Summer | Fog | 73.40 | 84.79 | 81.03 | 75.86 | 58.91 | 39.53 | 32.29 | 29.66 | 30.06 | 25.09 | 60.46 | 34.45 | **44.61** |
| Supervised | Fog | Fog | 98.10 | 94.71 | 97.70 | 92.32 | 96.05 | 80.53 | 57.03 | 92.72 | 51.03 | 85.27 | 86.87 | 64.99 | 83.11 |
| Before Adapt | Summer | Night | 46.12 | 63.16 | 77.78 | 69.77 | 36.72 | 55.25 | 35.04 | 57.84 | 30.72 | 36.32 | 48.43 | 27.85 | **41.79** |
| After Adapt | Summer | Night | 20.03 | 66.89 | 84.44 | 76.97 | 54.03 | 60.05 | 36.06 | 50.96 | 31.51 | 39.35 | 61.23 | 28.16 | **43.55** |
| Supervised | Night | Night | 98.27 | 94.71 | 97.90 | 91.60 | 95.03 | 81.21 | 51.56 | 96.54 | 46.85 | 84.31 | 84.88 | 49.93 | 81.06 |
| Supervised | Sunset | Sunset | 97.24 | 94.31 | 97.43 | 92.45 | 96.07 | 83.56 | 53.81 | 92.75 | 53.03 | 80.25 | 85.46 | 53.20 | 81.63 |
| Before Adapt | Sunset | Rain | 0.00 | 42.65 | 61.79 | 13.68 | 14.01 | 18.31 | 3.94 | 16.17 | 6.21 | 9.11 | 33.44 | 6.02 | **16.09** |
| After Adapt | Sunset | Rain | 0.00 | 45.60 | 58.21 | 19.03 | 32.18 | 31.88 | 4.01 | 22.03 | 7.04 | 18.57 | 32.44 | 9.57 | **23.38** |
| Supervised | Rain | Rain | 97.62 | 93.57 | 97.53 | 89.58 | 95.49 | 78.27 | 36.34 | 92.82 | 43.66 | 83.04 | 83.96 | 56.15 | 79.00 |
| Before Adapt | Sunset | Fog | 0.02 | 56.61 | 81.89 | 52.94 | 5.26 | 41.07 | 17.61 | 20.19 | 21.46 | 28.59 | 50.05 | 30.03 | **28.98** |
| After Adapt | Sunset | Fog | 0.00 | 64.33 | 93.76 | 80.09 | 73.40 | 59.71 | 24.96 | 61.82 | 27.28 | 46.62 | 72.08 | 36.65 | **49.29** |
| Supervised | Fog | Fog | 98.10 | 94.71 | 97.70 | 92.32 | 96.05 | 80.53 | 57.03 | 92.72 | 51.03 | 85.27 | 86.87 | 64.99 | 83.11 |
| Before Adapt | Sunset | Night | 0.00 | 62.45 | 76.01 | 63.19 | 29.69 | 45.92 | 31.34 | 39.19 | 27.30 | 22.51 | 57.39 | 19.83 | **33.92** |
| After Adapt | Sunset | Night | 0.00 | 59.26 | 81.42 | 65.10 | 59.31 | 63.17 | 31.85 | 63.83 | 25.05 | 32.69 | 43.83 | 25.84 | **39.38** |
| Supervised | Night | Night | 98.27 | 94.71 | 97.90 | 91.60 | 95.03 | 81.21 | 51.56 | 96.54 | 46.85 | 84.31 | 84.88 | 49.93 | 81.06 |

Table 2. **Adaptation across weathers on CARLA.** We study the cross weather performance using dataset from CARLA simulator. We report quantitative of supervising learning and comparisons of performance before and after adaptation for training on one weather and evaluating on another unannotated weather.

| Method | Source | Target | None | Buildings | Fences | Other | Pedestrians | Poles | Roadlines | Roads | Sidewalks | Vegetation | Vehicles | Walls | Traffic signs | mIoU |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Supervised | Clearnoon | Clearnoon | 93.51 | 85.43 | 50.36 | 53.77 | 14.96 | 33.26 | 77.43 | 97.70 | 89.76 | 79.91 | 71.81 | 67.83 | 51.66 | 66.72 |
| Before Adapt | Clearnoon | Wetnoon | 93.25 | 83.38 | 35.10 | 38.81 | 12.50 | 14.83 | 73.34 | 93.61 | 81.40 | 72.77 | 57.47 | 61.44 | 56.94 | **59.60** |
| After Adapt | Clearnoon | Wetnoon | 93.25 | 82.34 | 37.18 | 39.63 | 10.17 | 19.26 | 72.15 | 93.42 | 81.25 | 73.91 | 60.44 | 62.27 | 58.62 | **60.30** |
| Supervised | Wetnoon | Wetnoon | 93.93 | 87.23 | 43.42 | 48.64 | 4.85 | 43.16 | 78.97 | 98.04 | 89.91 | 80.67 | 73.34 | 66.55 | 78.12 | 68.22 |

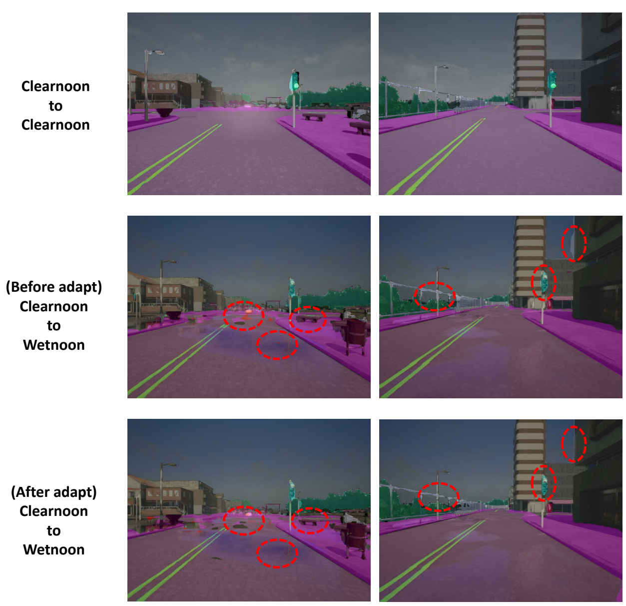


Figure 8. Visualization of CARLA experiment.

tation, as shown in Table 1. When the source domain is summer, On average over three weather,our global alignment method contributes 2.24% mIoU gain. Most of class show improve after adaptation, including sidewalk, fence, vegetation, traffic sign, pedestrian and traffic light. Interesting, when the source domain is sunset, on average over three

Table 3. **Adaptation across weathers on MVD.** We study the cross weather performance using real images from MVD dataset. We report quantitative of supervising learning and comparisons of performance before and after adaptation for training on one weather and evaluating on another unannotated weather.

| Method | Source | Target | mIoU |
|---|---|---|---|
| Supervised | Sunny | Sunny | 20.62 |
| Before Adapt | Sunny | Cloudy | 20.02 |
| After Adapt | Sunny | Cloudy | 16.93 |
| Before Adapt | Cloudy | Sunny | 15.83 |
| After Adapt | Cloudy | Sunny | 15.31 |
| Supervised | Cloudy | Cloudy | 19.31 |

weather, our global alignment method contributes 11.02% mIoU gain, showing significant improvement. We also can find most of class show improve after adaptation. Furthermore, some typical example of visualization results in Figure 7 also show that the improved segmentation adaptation from "before adaptation" to "after adaptation", demonstrating that our method is able to overcome this large appearance shift from different weather conditions.

**Cross Weathers on CARLA dataset** We perform an experiment using datasets from CARLA simulator for autonomous driving systems to demonstrate that our method could be generally applied to different datasets. The Table 2 show that the quantitative comparisons of performance supervising learning,before and after adaptation,using dataset from CARLA. On average we get 0.7% mIoU improvement for adaptation from clearnoon to wetnooon and find that our adaptation method provides higher mIoU for 7/13 object categories. This results prove that our propose method could be applied to different datasets and also shows improvement. In addition, we also present some typical example of visualization results in Figure 8, showing that the domain shift decreasing from "before adaptation" to "after adaptation". These results confirm that the effectiveness of our proposed domain adaptation method in different datasets.

### 5.4. Experiment on Real Images

In order to examine our method could be applied to the real image, we perform experiment in the MVD dataset. The quantitative comparisons of performance supervised learning, before and after adaptation is reported in Table 3. The results shows that our method does not improve the performance after adaptation. We can observe that even the supervised training show the poor results, resulting in model would not have good performance after adaptation. We suspect the reason is that there are too many classes in MVD dataset comparing to datasets from SYNTHIA and CARLA. Thus, it is necessary to modify our model to apply in MVD dataset.

## 6. Conclusion

Reasoning the semantic meaning of road scene is essential for UAV systems to plan how to act properly. However, the visual observation of road scene differs a lot under different weather conditions, which may cause self-driving system drastically fail. From our preliminary experiment, we found that semantic segmentation model degrades a lot when applying to different weather conditions. Therefore, in this project, we aim at learning unsupervised domain adaptation on road scene segmentation under different weather conditions, which is less studied in literature. We train semantic segmentation on a certain weather, and adopt unsupervised adversarial training to transfer the segmentation model to the target weather condition. First, we make use of synthetic dash-cam data from SYNTHIA datasets, to explore the domain shift of cross weather adaptation. Next, in order to examine our method could be applied to the real image, we conduct experiment in the MVD dataset. Finally, we perform an experiment using datasets from CARLA simulator for autonomous driving systems to prove that our method could be generally applied to different datasets for future applications. We show by experiment that our proposed method effectively align cross-weather data in feature space, and successfully adapt segmentation model to different weather conditions.

## References


[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *CoRR*, abs/1511.00561, 2015. 2

[2] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *CoRR*, abs/1606.00915, 2016. 2

[3] Y. Chen, W. Chen, Y. Chen, B. Tsai, Y. F. Wang, and M. Sun. No more discrimination: Cross city adaptation of road scene segmenters. *CoRR*, abs/1704.08509, 2017. 1, 3

[4] Y. Chen, W. Li, and L. V. Gool. ROAD: reality oriented adaptation for semantic segmentation of urban scenes. *CoRR*, abs/1711.11556, 2017. 1, 3

[5] J. Hoffman, D. Wang, F. Yu, and T. Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *CoRR*, abs/1612.02649, 2016. 1, 2, 3

[6] S.-W. Huang, C.-T. Lin, S.-P. Chen, Y.-Y. Wu, P.-H. Hsu, and S.-H. Lai. Auggan: Cross domain adaptation with gan-based data augmentation. In *The European Conference on Computer Vision (ECCV)*, September 2018. 3

[7] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038, 2014. 2, 4

[8] G. Neuhold, T. Ollmann, S. R. Bul, and P. Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. pages 5000–5009, Oct 2017. 5

[9] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015. 2

[10] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. pages 3234–3243, June 2016. 5

[11] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 4

[12] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. 2017. 2