# BoMuDANet: Unsupervised Adaptation for Visual Scene Understanding in Unstructured Driving Environments

Divya Kothandaraman, Rohan Chandra, Dinesh Manocha
University of Maryland, College Park

## Abstract

*We present an unsupervised adaptation approach for visual scene understanding in unstructured traffic environments. Our method is designed for unstructured real-world scenarios with dense and heterogeneous traffic consisting of cars, trucks, two-and three-wheelers, and pedestrians. We describe a new semantic segmentation technique based on unsupervised domain adaptation (DA), that can identify the class or category of each region in RGB images or videos. We also present a novel self-training algorithm for multi-source DA that improves the accuracy. Our overall approach is a deep learning-based technique and consists of an unsupervised neural network that achieves 87.18% accuracy on the challenging India Driving Dataset. Our method works well on roads that may not be well-marked or may include dirt, unidentifiable debris, potholes, etc. A key aspect of our approach is that it can also identify objects that are encountered by the model for the fist time during the testing phase. We compare our method against the state-of-the-art methods and show an improvement of* $5.17\% - 42.9\%$. *Furthermore, we also conduct user studies that qualitatively validate the improvements in visual scene understanding of unstructured driving environments.* [1]

## 1. Introduction

Visual scene understanding is a key component of perception systems in autonomous vehicles (AVs) that deals with tracking, prediction, object detection, classification, localization, and semantic segmentation [49, 34]. These systems are responsible for understanding or interpreting the environment for safe and efficient navigation and collision avoidance. There has been considerable work on developing systems that are now deployed in the current generation of AV technologies [2]. However, most of the scene understanding algorithms and systems have been developed for highly controlled or structured environments. This includes sparse or homogeneous traffic, well-structured roads with clear lane marking, absence of debris, potholes or unidentifiable objects etc. Many times, the AVs need to operate in unstructured scenarios that consist of heterogeneous traffic with cars, trucks, bicycles or pedestrians and there is less adherence to traffic rules or regulations. Furthermore, roads are not well-marked or may include dirt or unidentifiable debris. There are still many challenges in terms of developing robust perception systems for such unstructured environments.

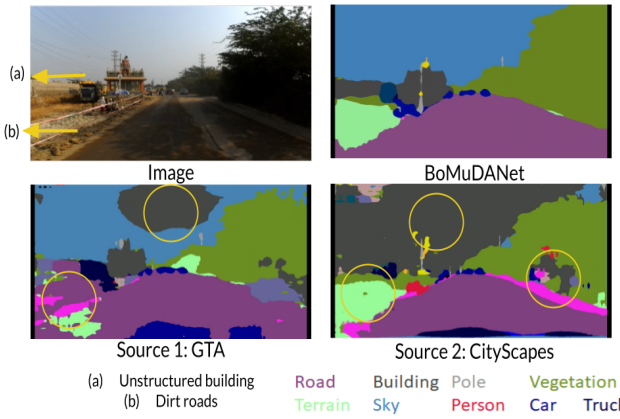


Figure 1: We present a novel unsupervised deep learning-based approach called BoMuDANet for visual scene understanding in unconstrained and unstructured traffic environments. In this example, we demonstrate the benefits of BoMuDANet on images taken from the challenging IDD dataset. BoMuDANet accurately segments out dirt roads as terrains as well as a building, while preserving its shape. In contrast, the single source baselines (GTA/CityScapes) do not identify dirt and unstructured roads well, misclassify parts of sky as building, and fail to capture the shape of the unstructured building. BoMuDANet benefits from its ability to selectively distil information from various sources by iterative self-training, in addition to exploiting a chosen best source via domain adaptation.

Recent developments in deep learning [7, 49] have resulted in significant advances in visual scene understanding [14, 45] in structured traffic environments [48, 9]. However, they do not work well in unstructured or unconstrained environments like India Driving Dataset [40] (see Figure 1). This is mainly due to lack of good datasets that consists of unstructured traffic scenes or outlier objects. For example, current traffic datasets used for training may not have vehicles such as auto-rickshaws [5]. An approach that can over-

[1] Code and Video at https://github.com/divyakraman/BoMuDA-Boundless-Multi-Source-Domain-Adaptive-Segmentation-in-Unstructured-Environments

come these challenges is domain adaptation (DA) [39], a transfer learning technique that takes advantage of the availability of large scale annotated data in a different domain called the 'source' domain to perform a task on the 'target' domain, for which data is typically scarce. More specifically, DA can leverage many available large-scale structured traffic datasets such as CityScapes [9], Berkeley Deep Drive [48], GTA [31], and SynScapes[44] (source domains) to learn robust feature representations in unstructured environments.

**Main Contributions:** We present a new deep neural network called BoMuDANet for visual scene understanding in unstructured traffic environments. Our approach consists of a semantic segmentation technique based on unsupervised domain adaptation. BoMuDANet includes two novel components:

1. Unconstrained traffic environments are highly heterogeneous (wide range of object classes) [6]; consequently, using only one source dataset (single source DA) is not sufficient in terms of providing the network with adequate information for optimal performance on the complex target domain. BoMuDANet benefits from its ability to selectively extract relevant knowledge across different and widely available structured environment datasets [48, 9]. Moreover, we perform multi-source DA [53] by alternating between adaptation from a selected source and knowledge distillation from the remaining sources. Based on the classical EM algorithm in statistical pattern recognition, BoMuDANet performs repeated rounds of training, alternating between adaptation and distillation to improve performance in each step. We present the self-training algorithm in Section 3.3.

2. Unconstrained traffic scenes may contain objects that are typically non-existent in current structured environment source domain datasets. Our approach in BoMuDANet uses a simple pseudo-labeling strategy (Section 3.4) for handling unknown objects encountered for the first time during the testing phase. The final probability predictions of the self-training algorithm are directly used to assign proxy labels to unknown object classes depending on their similarity to known objects classes in the training dataset. Our approach helps BoMuDANet detect new objects that are common in unstructured driving environments.

We have evaluated our approach extensively using the Indian Driving Dataset (IDD), CityScapes, Berkeley DeepDrive, GTA V, and the Synscapes datasets. In unstructured environments (IDD as the target dataset), we show that our unsupervised approach outperforms other unsupervised SOTA benchmarks by $5.17\% - 42.9\%$. In structured environments (CityScapes as the target dataset), we show that our method outperforms other multi-source DA methods by $12.70\% - 90.13\%$. We have performed extensive ablation experiments to highlight the benefits of our approach.

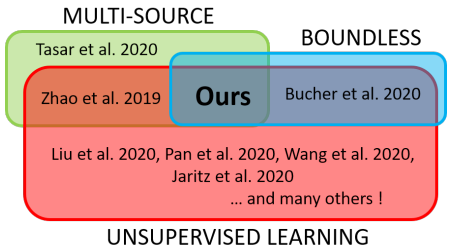


Figure 2: Extension to SOTA in domain adaptive semantic segmentation. Our approach is the first method to *simultaneously* perform unsupervised multi-source boundless DA segmentation and can handle unstructured traffic environments.

Moreover, we also perform a user study to highlight the qualitative benefits of our approach. Overall, ours is the first unsupervised domain adaptation method for handling unstructured traffic environments.

## 2. Prior Work

There is considerable work in domain adaptation (DA) for semantic segmentation and other perception tasks. While a detailed review of these methods is not within the scope of this paper, we briefly mention related work.

### 2.1. Semantic Segmentation

Semantic segmentation is a pixel-level task, which involves assigning a label to each pixel in an image. The advent of deep learning has resulted in a many segmentation techniques for autonomous driving [13, 49, 7, 36, 10, 12]. However, these methods suffer from three issues: (i) the networks need to be trained in a supervised manner, thus there is a demand for large volumes of annotated data; (ii) current labeled datasets are limited to structured environments; (ii) current learning methods do not scale well to unstructured environments.

### 2.2. Unsupervised Domain Adaptation

Domain adaptive semantic segmentation has been explored under three different machine learning paradigms that differ based on the underlying learning approach. At one end of the spectrum, fully supervised [1] methods achieve higher accuracy on average, but are limited by the availability of annotated data. On the other end of the spectrum, unsupervised methods'[28, 41, 38, 23, 28, 55, 50, 15, 43, 46] benefit from the lack of dependence on any training data, but are outperformed by fully supervised or semi-supervised methods. Semi-supervised approaches [18, 20, 32, 30] form a middle ground between the two paradigms. While there has been some work on using pseudo labels [4, 19] for training DA models, they do not scale well in the presence of multiple sources.

### 2.3. Open-Set and Boundless DA

If the set of labels in the source is equal to the set of labels in the target, then this type of DA is known as *closed-set DA*. On the other hand, if the target domain contains

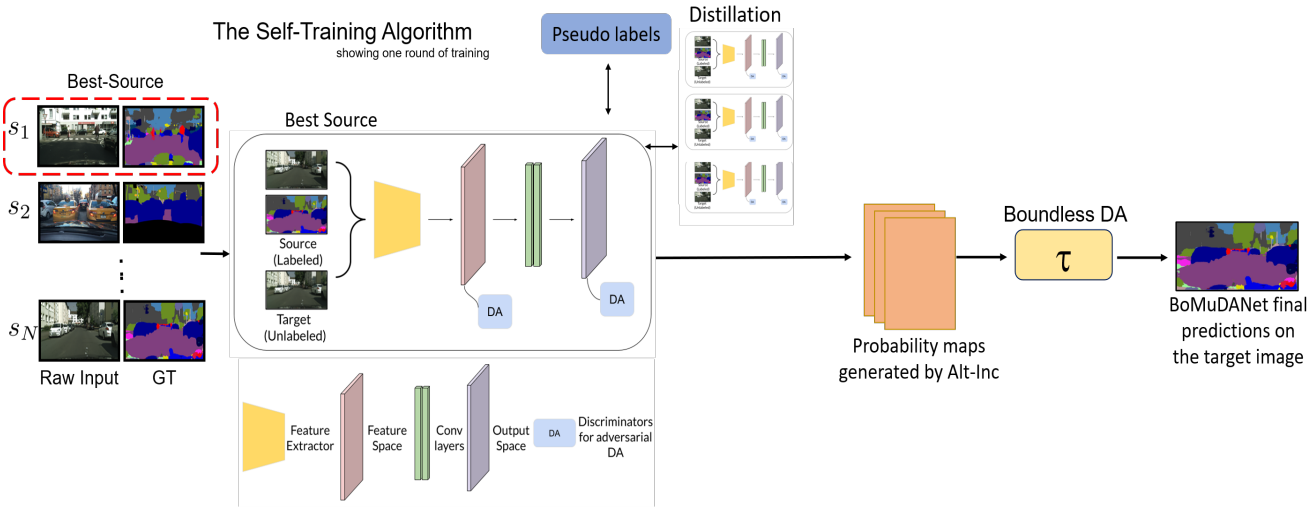


Figure 3: **Overview of BoMuDANet:** The input consists of $N$ sources $(s_1, s_2, \ldots, s_N)$, from which the best-source is selected by the self-training algorithm (Section 3.3). The self-training algorithm proceeds in an unsupervised fashion to generate the final set of pseudo-labels that are used to recognize out-of-distribution objects (Section 3.4). The final output consists of the segmentation map of an image in the target domain.

additional class labels that are not present in the source domain, then this type of DA is called *open-set DA*. In open-set DA, the additional class labels in the target domain that do not belong to the source domain are labeled as an "unknown" class [37]. While open-set DA has been proposed for object detection and classification [29, 33, 24], they don't extend well for pixel-level tasks like semantic segmentation.

An extension to open-set DA is boundless DA, where the extra classes present in the target domain are explicitly labeled. Boundless DA has been recently studied by [4] for semantic segmentation, where the authors successfully classify open-set classes, but at the cost of degraded accuracy on the closed-set categories.

### 2.4. Multi-Source Domain Adaptation

While multi-source DA has been extensively studied in the context of other perception tasks like object recognition and classification [11, 22, 54, 42, 47], it has not been explored in detail for semantic segmentation [53, 51]. Prior approaches in multi-source DA suffer from heavy overhead in terms of the requirement of data from all sources at every point of training. In contrast, BoMuDA requires only the pre-trained models, along with data from a chosen source 'best source' domain.

We present the first method for unsupervised multi-source boundless domain adaptive semantic segmentation (See Figure 2). However, our approach can be generally applied towards domain adaptation in different perception tasks such as object recognition. This is a part of our future work.

## 3. BoMuDANet

In this section, we formally specify our problem, introduce the notation and present details of our neural network used for visual scene understanding in unstructured traffic environments.

### 3.1. Problem Setup and Notation

Given an RGB image or video of unconstrained traffic selected from the target domain at test time, our goal is to identify the correct object class label of each pixel. In the training phase, we are provided with a set $\mathcal{S}$ of $N$ source domains, in which each source domain is represented as $S_i$, where $i = 1, 2, \ldots, N$, and one target domain $T$. The set of all categories in the target domain is denoted by $\mathcal{C}_T$, while the set of all categories in the $i^{\text{th}}$ source domain is denoted by $\mathcal{C}_i$. In the boundless DA setting, the target domain may consist of open-set categories *i.e.* classes that are not present in any of the source domains. More formally, $\mathcal{C}_T \setminus \{\cup_i \mathcal{C}_i\} \neq \emptyset$.

The output probability map[2] for an input image belonging to the $i^{\text{th}}$ source domain is denoted as $P_i \in \mathbb{R}^{|\mathcal{C}_i| \times h \times w}$, while the ground truth label for the same image is denoted by $y_i \in \mathbb{R}^{h \times w}$. In the unsupervised DA setting, the ground-truth labels for target domain images are not available. We present BoMuDANet in Section 3.2.

### 3.2. Overview

Our method, BoMuDANet (Figure 3), trains a deep neural network using a novel self-training algorithm (Section 3.3). The input consists of images and their corresponding labels from multiple source domains and images from

[2]Each value in this map is the probability of the pixel, in the corresponding location in the input image, belonging to class $\mathcal{C}_i$.

the target domain. The self-training algorithm generates probability maps corresponding to the target domain image.

### 3.3. A Self-Training Algorithm for Multi-Source DA

The main challenge with unsupervised multi-source domain adaptation is in setting up a cost function [52] for training the deep neural network (DNN). This is due to two reasons: $(i)$ absence of target domain labels and $(ii)$ variations between different source domains, and each source and target domain. In the proposed approach, we use the idea of "pseudo" labels to act in place of the missing target domain labels. The pseudo labels are the class predictions from the 'best source' single-source DA model, which is explained below in "Initialization". These pseudo labels, along with the pre-trained single-source DA models from the remaining sources are used for training the deep neural network with an improvised cost function.

In concurrence with the notion of self-training, we observed that repeated re-training of the deep neural network with an enriched cost function each time leads to a significant boost in accuracy. This is because the model weights are optimized after each round of training, which in turn optimizes the pseudo labels, leading to an increasingly accurate cost function to be used in the next round of training[3]. These two optimizations occur in an alternating manner along with domain adaptation from best source and multi-source distillation, incrementally improving the accuracy.

The motivation behind the self-training algorithm comes from the Expectation-Maximization (EM) algorithm [27], a classical unsupervised learning algorithm in statistical pattern recognition. The EM algorithm consists of two alternating steps– the E step and the M step. The E step sets up a cost function from observed data, while the M step finds the model parameters that minimize the cost function. Our self-training algorithm mimics the principles behind the EM algorithm.

To setup the cost function for the first iteration, the self-training algorithm selects the best-performing source from the $N$ source domains to generates pseudo-labels that serve as a proxy for the missing target domain labels. This best-performing source dataset is termed as the "Best-Source". The inputs to the neural network are images and GT from the "Best-Source", pre-trained single-source DA models for all the sources, images from the target domain and the pseudo labels. In summary, the self-training algorithm (Figure 3) trains the network as follows:

1. *Initialize* $\leftarrow$ Best-Source model.
2. *Perform the following in an alternating manner:*
   - Use pseudo labels from previous round of training to set up a cost function for BoMuDANet.
   - Use this cost function along with the remaining $N$–1 pre-trained single-source DA models to train BoMuDANet in an end-to-end manner till convergence.

[3]We provide evidence of the benefits of repeated re-training in the supplementary material.

We now describe each step in detail.

#### 3.3.1 Initializing the Best Source Model

We begin by training single-source DA models using each source dataset, and the target dataset, using an adversarial DA paradigm [38]. The single-source domain discriminators (binary classifiers, see architecture below) characterize how indistinguishable the target domain is from each source domain. The output of the discriminators averaged over all target images characterizes the dissimilarity between each source domain and the target domain. The source domain with the least dissimilarity is selected as the "best source". The deep neural network (DNN) used to train the best source-target pair is termed the "Best-Source" model.

**Architecture:** Consistent with adversarial domain adaptation [38], our network consists of a DNN for semantic segmentation, and domain discriminators. The backbone of the DNN consists of SOTA architectures such as VGG-16 [35], Dilated Residual Network [49], or DeepLab [7] (we experiment with different backbones in Section 4). Domain discriminators are neural networks that aim to distinguish whether the predicted segmentation map is from the source or target.

**Training:** The inputs to this model consist of raw images and GT from the best source domain, raw images from the target domain $T$, and the pseudo labels. The model weights are initialized with parameters corresponding to the Best-Source baseline obtained in the initialization step. The cost function consists of a domain adaptation loss formulation to adapt from the best-source, a knowledge distillation loss formulation to selectively distil relevant information from the remaining sources, and a self-training step that utilizes the pseudo labels. We now describe the three loss functions that are used to train the network:

- **The supervised loss function** ($\mathcal{L}_{\text{sup}}$): This is the standard cross entropy supervised loss function that is used to minimize the distance between the probability map outputs and the ground truth labels.

$$\mathcal{L}_{\text{sup}} = -\sum_{h,w}\sum_{c\in\mathcal{C}_{\text{bs}}} y_{\text{bs}}\log(P), \tag{1}$$

  where $c$ denotes the object category, $h, w$ denote the height and width of the input images, and $P \in \mathbb{R}^{|\mathcal{C}_{\text{bs}}|\times h\times w}$ is the output of the model on source domain images.

- **The unsupervised loss function** ($\mathcal{L}_{\text{unsup}}$): For each target image, we use the trained model from the previous iteration of self-training to generate pseudo labels for self-training. The pseudo labels are generated using the probability map predictions, $P \in \mathbb{R}^{|\mathcal{C}|\times h\times w}$.

More formally,

$$y_{\text{pseudo}} = \arg\max_{c \in \mathcal{C}} \text{Softmax}(P). \quad (2)$$

The pseudo-label $y_{\text{pseudo}}$ is used in the unsupervised cross entropy loss function, $\mathcal{L}_{\text{unsup}}$, as follows,

$$\mathcal{L}_{\text{unsup}} = -\sum_{h,w}\sum_{c \in \mathcal{C}} y_{\text{pseudo}} \log(P). \quad (3)$$

- **Multi-source distillation** ($\mathcal{L}_{\text{distill}}$): From each of the single source DA networks, we generate their corresponding target domain probability maps $P_i, i \in [N]$. To selectively impart relevant knowledge from various sources, the target domain predictions of BoMuDANet are distilled using a weighted combination of KL divergence [25] loss terms corresponding to each of the single-source DA predictions. Kl divergence aligns probability distributions and a weighted combination of KL divergence from multiple sources aids in selective extraction of relevant knowledge. The weights $(w_i)$ are determined by the dissimilarity between each source-target pair (see initialization step above).

$$\mathcal{L}_{\text{distil}} = \sum_i w_i \times KL(P_{\text{bs}} || P_i). \quad (4)$$

The three loss functions are combined as follows:

$$\mathcal{L}_{\text{overall}} = \lambda_{\text{sup}}\mathcal{L}_{\text{sup}} + \lambda_{\text{unsup}}\mathcal{L}_{\text{unsup}} + \lambda_{\text{distil}}\mathcal{L}_{\text{distil}}, \quad (5)$$

where $\lambda_{\text{sup}}$, $\lambda_{\text{unsup}}$, $\lambda_{\text{distil}}$ denote the hyperparameters for the respective loss terms. The domain discriminators are trained in an adversarial [38] fashion.

### 3.4. Boundless Domain Adaptation

We present a new method for performing Boundless DA *i.e.* to label categories that exist in the target dataset, but not in any of the source datasets ("open-set" or "private" or "unknown" categories). Categories that are common to both the source and the target domains are called "closed-set" or "shared" or "known" categories. The key assumption in our solution is that the open-set categories are physically similar to the closed-set categories. For instance, open-set categories such as auto-rickshaws are similar to vehicles like cars and vans. CityScapes [9] provides a definition for grouping semantically similar classes in autonomous driving environments. Classes that belong to the same high-level category will have feature maps that are semantically similar, and vice versa. This assumption is mild and is commonly made in many zero-shot learning strategies [3].

The underlying idea behind training our approach on open-set classes is to generate the corresponding pseudo-labels from the labels of the physically similar closed-set categories. More formally, let $y_{\text{ST}} \in \mathbb{R}^{h \times w}$ be the final labels obtained using Equation 2 from the self-training algorithm. Further, let $o \in \mathcal{O}$ denote an open-set class from the set of open-set classes, $\mathcal{O}$, and $\mathcal{C}_o$ denote the set of closed-set classes that are physically similar to $o$. We apply thresholding on $y_{\text{ST}}$ such that pixels with softmax scores lower than a threshold $\tau$ for a physically-similar closed-set class are re-labeled as the open-set class. More formally, let $\hat{y}_{\text{pseudo}}$ denote the labels after thresholding, then $\hat{y}_{\text{pseudo}}$ is computed using,

$$\hat{y}_{\text{pseudo}} = \mathcal{T} y_{\text{ST}} \quad (6)$$

where $\mathcal{T}(\cdot)$ is a pixel-level thresholding operator. If $l_{ab}$ denotes the class label of a pixel in the $a^{\text{th}}$ row and $b^{\text{th}}$ column with confidence score $c_{ab}$ (maximum probability value over all classes, as determined by the output probability map of the self-training algorithm), then the threshold operator at $(a, b)$ is defined as,

$$\mathcal{T}(a,b) = \begin{cases} l_{ab} \leftarrow o & c_{ab} \leq \tau \text{ and } l_{ab} \in \mathcal{C}_o \\ l_{ab} & \text{otherwise} \end{cases}$$

An alternative to the thresholding operator is to use the KL divergence metric [25] to measure the similarity between open-set and closed-set object classes. We empirically observe, however, via an ablation study that thresholding in fact outperforms using the KL divergence metric (See Table 2).

## 4. Experiments and Results

We will make all code publicly available. We defer the technical implementation details of the training routine including hyper-parameter selection to the supplementary material.

### 4.1. Datasets and Evaluation Protocol

We use five datasets - GTA5 [31], SynScapes (SC) [44], CityScapes (CS) [9], India Driving Dataset (IDD) [40] and Berkeley Deep Drive (BDD) [48]. GTA5 and SC contain synthetic simulated traffic videos while CS and BDD consist of real-world traffic in Europe and the USA, respectively. IDD consists of dense and unconstrained traffic conditions and heterogeneous road agents (e.g. autorickshaws) unobserved in any of the source domains. In addition to containing new objects, the pixel count (per class) in IDD is $5{-}10\times$ that of CS

We evaluate our models on the validation set images of the target domain, using the standard segmentation metrics [26]: Intersection over Union (IoU) and pixel-wise accuracy.

### 4.2. Results

**Main Results on IDD (Table 1):** We present results of three sets of experiments using IDD as the target dataset in Table 1. In each experiment, we compare BoMuDANet with single-source baseline models using the BDD, CS, SC, and GTA datasets, with the BDD dataset selected as the Best-Source. Note that we do not compare with a combination of single source datasets as combining multiple sources and treating them as a single source for DA has been shown to be ineffective [52]. We compare with SOTA multi-source

| Model | Experiment | mIoU (↑) | mAcc (↑) | Road | SW | Bldg | Wall | Fnc | Pole | Lt | Sign | Veg | Trn | Sky | Ped | Rdr | Car | Trk | Bus | Mb | Bike |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I. CS, BDD, GTA ⟶ IDD (Baseline: [38]) | | | | | | | | | | | | | | | | | | | | | |
| Baselines | CS⟶IDD | 24.43 | 65.23 | 82.46 | 22.55 | 25.93 | 13.22 | 9.30 | 15.26 | 1.92 | 19.02 | 75.16 | 20.41 | 29.54 | 31.37 | **8.12** | 49.81 | 8.53 | 10.41 | **10.29** | 6.55 |
| | GTA⟶IDD | 26.74 | 75.40 | 79.83 | 9.54 | 44.12 | **16.58** | 12.16 | 17.59 | 0.85 | 14.35 | 65.36 | 18.20 | 82.61 | 22.90 | 6.56 | 41.53 | 24.13 | 15.40 | 9.02 | 0.76 |
| | BDD⟶IDD | 35.75 | **85.65** | 93.33 | 27.17 | 59.77 | 13.18 | 15.56 | **21.03** | **3.65** | 29.93 | 80.52 | 33.21 | 93.64 | 30.62 | 5.59 | 53.03 | 38.34 | 32.24 | 6.46 | 6.27 |
| BoMuDANet | Multi-source | **37.66** | **86.50** | **94.02** | **31.89** | **61.79** | **15.51** | **16.89** | **20.61** | 2.73 | **35.43** | **81.75** | **36.52** | **94.16** | **32.12** | 4.67 | **54.74** | **42.64** | **38.61** | 5.42 | **8.51** |
| II. SC, BDD, GTA ⟶ IDD (Baseline: [38]) | | | | | | | | | | | | | | | | | | | | | |
| Baselines | SC ⟶IDD | 31.55 | 83.04 | 92.46 | 21.25 | 52.59 | 4.61 | 7.87 | 17.02 | 2.73 | 12.60 | 77.52 | 4.43 | 92.38 | 31.54 | **23.32** | **66.59** | 4.09 | 18.35 | **27.27** | **11.25** |
| | GTA⟶IDD | 26.74 | 75.40 | 79.83 | 9.54 | 44.12 | **16.58** | 12.16 | 17.59 | 0.85 | 14.35 | 65.36 | 18.20 | 82.61 | 22.90 | 6.56 | 41.53 | 24.13 | 15.40 | 9.02 | 0.76 |
| | BDD⟶IDD | 35.75 | 85.65 | 93.33 | 27.17 | 59.77 | 13.18 | 15.56 | 21.03 | **3.65** | 29.93 | 80.52 | 33.21 | 93.64 | 30.62 | 5.59 | 53.03 | 38.34 | 32.24 | 6.46 | 6.27 |
| BoMuDANet | Multi-source | **36.93** | **86.30** | **93.82** | **30.53** | **61.13** | **13.34** | **16.43** | **21.21** | **3.57** | **34.90** | **81.64** | **34.54** | **94.19** | **31.70** | 4.64 | **53.48** | **40.77** | **35.54** | 5.68 | **7.64** |
| III. CS, BDD, GTA ⟶ IDD (Baseline: [41]) | | | | | | | | | | | | | | | | | | | | | |
| Baselines | CS ⟶ IDD | 38.53 | 86.68 | 93.67 | 27.08 | **64.62** | **25.89** | 17.80 | 23.39 | 4.18 | 31.29 | **83.06** | 29.83 | 94.22 | 32.28 | 11.18 | 61.68 | 39.86 | 33.32 | 12.08 | 8.23 |
| | GTA ⟶ IDD | 35.85 | 84.64 | 89.96 | 14.06 | 61.14 | 22.24 | 20.10 | 19.17 | 4.34 | 19.88 | 77.15 | 28.84 | 92.14 | 27.03 | **11.98** | **62.87** | 41.04 | 34.67 | 13.10 | 5.74 |
| | BDD ⟶ IDD | 38.29 | 86.74 | **93.80** | **33.33** | 62.57 | 14.94 | 15.35 | **23.66** | 3.80 | **31.95** | 81.72 | 34.47 | **94.26** | **33.00** | 8.71 | 57.11 | 42.87 | 39.16 | 9.41 | 9.22 |
| BoMuDANet | Multi-source | **39.23** | **87.18** | 93.18 | **29.97** | **63.46** | **24.18** | **20.97** | **19.18** | **4.56** | 25.64 | **81.99** | **35.39** | 94.19 | 30.06 | **11.23** | **62.01** | **46.65** | **39.30** | **13.39** | **10.87** |

Table 1: **Main Results:** We evaluate BoMuDANet on IDD using CityScapes (CS), Berkeley Deep Drive (BDD), SynScapes (SC), and GTA as sources. Higher (↑) mIoU and mAcc indicates direction of better performance. **Bold** indicates best while **blue** indicates second-best. Experiments I and II differ with respect to the sources, while experiment III differs with respect to the baseline used. **Conclusion:** Our unsupervised multi-source self-training algorithm outperforms the single-source baselines by $3.3\% - 54.15\%$.

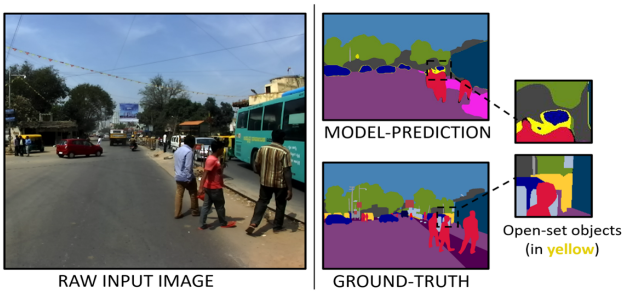


Figure 4: In this example, we demonstrate the benefits of BoMuDANet on an image from the IDD dataset, depicting a mixture of challenging driving conditions. The top image in the second column shows the prediction of our model, and the bottom image shows the ground-truth. We observe that BoMuDANet accurately segments the autorickshaws (open-set object, a new type of vehicle - the third column zooms into the region containing the autorickshaw in the prediction and ground-truth), in addition to handling dense traffic and dirt roads.

DA methods in Section 4.3.

We perform the first experiment with two real datasets (CS, BDD) and one synthetic dataset (GTA5), and show an improvement of $1.91 - 13.23 (5.34\% - 54.15\%)$ mIoU points over the single-source baselines. In the second experiment, we replace the two real source datasets with two synthetic source datasets (SC, GTA) and one real dataset (BDD) with the BDD dataset as the Best Source, and show an improvement of $3.3\% - 38.1\%$ mIoU points over the single-source baselines. By comparing these two sets of experiments, we demonstrate that using multiple real-world datasets is more beneficial than using multiple synthetic datasets.

In the third experiment, we replace the AdaptSegNet [38] backbone with a stronger SOTA backbone ADVENT [41], and use LS GAN for adversarial training instead of Vanilla GAN, and achieve a higher mIoU of 39.23. This suggests that the performance of our approach will increase as newer robust backbone architectures are proposed. We also validate this hypothesis by conducting experiments on structured environments as the target domain.

| Experiment | mIoU (↑) | Car | Truck | Bus | **Auto** |
|---|---|---|---|---|---|
| CS, BDD, GTA ⟶ IDD, Baseline: [38] | | | | | |
| Pseudo labeling (PL) | 35.68 | 51.16 | 33.89 | 28.99 | **9.39** |
| PL + training | 35.72 | 52.18 | 33.93 | 31.65 | **9.38** |
| SC, BDD, GTA ⟶ IDD, Baseline: [38] | | | | | |
| Pseudo labeling (PL) | 34.60 | 48.36 | 30.78 | 20.82 | **9.68** |
| PL + training | 34.40 | 49.14 | 30.44 | 22.59 | **9.48** |
| CS, BDD, GTA ⟶ IDD, Baseline: [41] | | | | | |
| Pseudo labeling (PL) | 37.27 | 58.76 | 36.58 | 22.15 | **11.78** |
| PL + training | 37.09 | 58.63 | 36.65 | 24.26 | **11.85** |
| Ablation Experiments on IDD, Baseline: [38] | | | | | |
| KL Divergence | 35.43 | 52.12 | 33.99 | 31.46 | 9.29 |

Table 2: **Pseudo labeling strategy for boundless DA:** We show that pseudo labeling can provide semantic information abuot categories that do not belong to any source domain, for instance, **autorickshaws** (Auto) found in the IDD dataset (in **bold**). Moreover, pseudo labeling is simple in that the generated proxy labels do no need to be re-trained as there is no marked improvement in mIoU. We also show that thresholding outperforms KL divergence via an ablation study.

**Results for the Boundless Case:** In Table 2, we show the results for the proposed pseudo labeling strategy for boundless DA method. Note that the thresholding operator, $\tau$, is a tunable hyperparameter. A low value of $\tau$ will create a bias towards the private classes, a high value of $\tau$ will create a bias towards shared classes. A trade-off determines the optimal value of $\tau$ for best performance on both private and shared classes. Typically, tuning between $80\% - 90\%$ of max confidence score for the particular class works well.

The unsupervised loss function in Equation 3 can be used

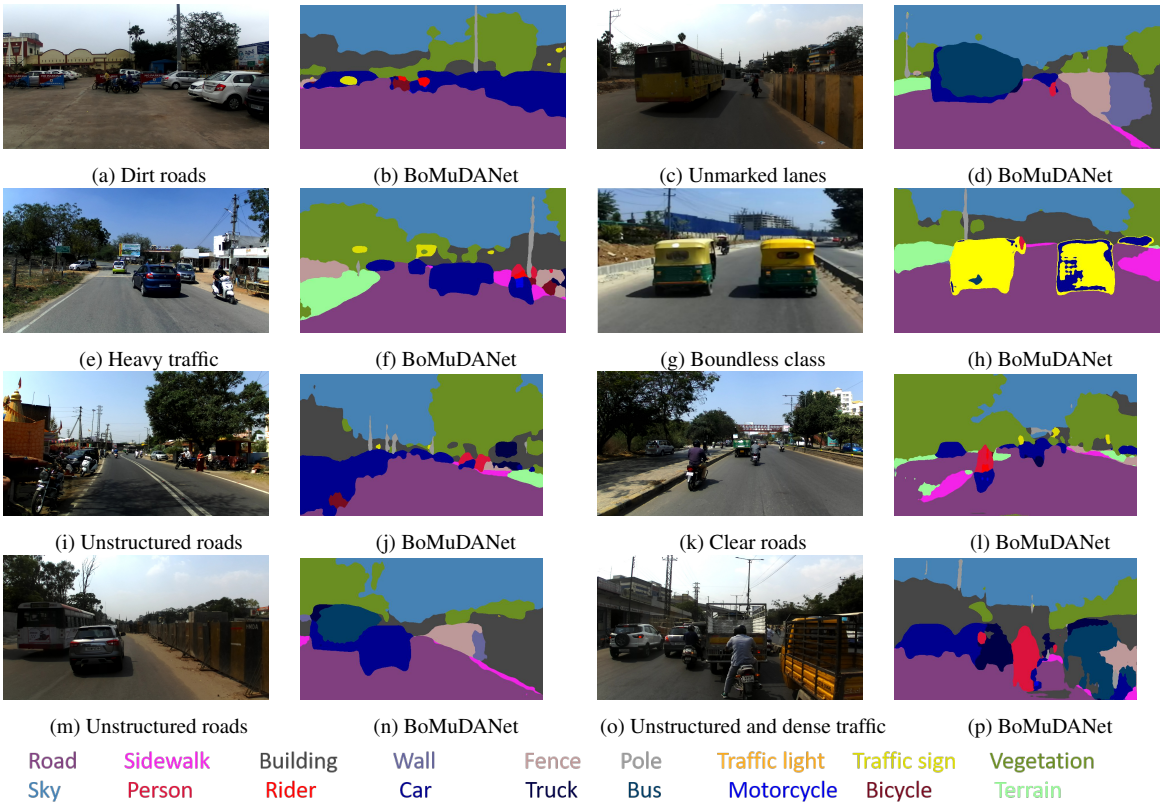


Figure 5: **Visual Results:** BoMuDANet works well in various unconstrained environments including unmarked lanes, dirt roads, heavy traffic, and boundless category objects (auto-rickshaws) and results in higher accuracy. Each color represents a different object as shown in the color scheme.

to retrain the network on $\hat{y}_{\text{pseudo}}$, along with $y_{\text{ST}}$ which acts as a regularizer. The first row in each experiment shows the results obtained by proposed strategy of pseudo-labeling while the second row shows the results obtained by re-training the generated pseudo labels. However, we found that the performance of the retrained model is very similar to original model, mitigating the need for costly retraining and therefore contributing to the simplicity of the proposed pseudo labeling strategy.

**Qualitative Results and Realtime Performance:** We present the qualitative results in Figure 5. Our method works well in environments that have dirt roads, absence of clear lane markings, multiple road objects and unstructured traffic. Figures 5g and 5h show that BoMuDANet can recognize auto-rickshaws (boundless category object) reasonably well.

We have included a video demonstration of BoMuDANet in realtime in 6 diverse traffic videos containing both unconstrained (IDD) as well as structured (CityScapes) environments, along with comparisons. BoMuDANet operates at 2 fps on IDD and 21 fps on CityScapes with a model size of 26.5 million parameters. We refer the reader to the supplementary video.

### 4.3. Comparsons with SOTA

**In Unstructured Environments (Table 3, On IDD):** In Table 3 (On IDD), we compare our approach against other unsupervised segmentation methods. ZS3Net [3] does zero shot semantic segmentation, while [4] (UDA) and [4] (Apt.) builds upon ZS3Net for domain adaptation. [4] (Comb.) refers to the combined approach for boundless unsupervised domain adaptation ("BUDA"). It can be clearly observed that our method surpasses all past unsupervised segmentation methods by 5.17% - 34.34% on shared classes, with a much smaller architecture (Table 3, model size) which is beneficial for practical autonomous driving real-time applications.

Our hypothesis for the superiority of BoMuDANet over BUDA is that the latter comprises performance on closed-set classes in order to achieve improved performance on open-set classes [3, 4]. In contrast, our method classifies open-set categories *without* sacrificing accuracy on closed-set categories (Table 2, Figure 5). The decreased performance of BUDA on "shared" classes could be due to decreased generalization capabilities of the model when trained on the new classes. We also outperform the semi-supervised method [17] by 42.9%, that uses ground-truth in 100 samples for supervision. [17] fails to acknowledge dif-

| On IDD | | | | |
|---|---|---|---|---|
| Method | Model | # Size(M↓) | mIoU(S↑) | mIoU(P↑) |
| [16] | ResNet-18 | **11.70** | 27.45 | NA |
| [3] | ResNet 101 | 44.50 | 29.20 | 7.90 |
| [4] (UDA) | ResNet 101 | 44.50 | 32.40 | 8.10 |
| [4] (Apt.) | ResNet 101 | 44.50 | 32.70 | 8.60 |
| [4] (Comb.) | ResNet 101 | 44.50 | 37.30 | **18.50** |
| **BoMuDANet** | **DRN-D-38** | **26.50** | **39.23** | **11.85** |

| On CS | | | | |
|---|---|---|---|---|
| Method | Model | Size(M↓) | mIoU(S↑) | # Sources |
| [8] | ResNet-101 | 44.50 | 39.40 | Single |
| [38] | ResNet-101 | 44.50 | 42.40 | Single |
| [41] | ResNet-101 | 44.50 | 43.10 | Single |
| [41] | ResNet-101 | 44.50 | 43.80 | Single |
| [28] | ResNet-101 | 44.50 | 46.30 | Single |
| [21] | ResNet-101 | 44.50 | 49.90 | Single |
| [51] | VGG-16 | 138.00 | 29.40 | Multi |
| [53] | VGG-16 | 138.00 | 41.40 | Multi |
| BoMuDANet | **DRN-D-38** | **26.50** | 44.63 | Multi |
| BoMuDANet | ResNet-101 [41] | 44.50 | 49.59 | Multi |
| **BoMuDANet** | ResNet-101 [21] | 44.50 | **55.90** | Multi |

Table 3: **Comparison with SOTA:** We compare with the SOTA in both unstructured (IDD) as well as structured (CS) traffic. Higher (↑) mIoU and mAcc indicates direction of better performance. **Bold** indicates best while **blue** indicates second-best. mIoU(S) and mIoU(P) denote the performance on shared/known and private/unknown classes, respectively. **Conclusion:** Our model is SOTA on IDD by $5.17\% - 42.9\%$ and on CS in the multi-source setting by $12.70\% - 90.13\%$, with a reduction in model size by upto to $5.2\times$.

ferences between various domains, which leads to a degradation in performance.

**In Structured Environments (Table 3, On CS):** We additionally benchmark BoMuDANet in structured environments, using CS as the target domain and BDD, IDD and GTA as the source domains. Our method is SOTA in the multi-source setting by at least $12.70\%$–$90.13\%$ with a reduction in model size by upto $5.2\times$. Methods with ResNet-101 backbone have an inference time of $156.44$ ms, and models with DRN-D-38 backbone have an inference time of 51.58 ms. On CS, BoMuDA outperforms the corresponding single-source DA baselines by $2.5\% - 21.2\%$ respectively. Furthermore, stronger backbones will help our model benefit accordingly (Table 1 I and III; and second half of Table 3). Further, comparison of our network (with corresponding backbones) against single-source baselines reveals that our model is the SOTA (Table 1, second half of Table 3).

The core step in the approach of [53] is the use of the CycleGAN [56], which uses images and ground truth from all source domains at every training step. Our multi-source approach, in contrast, is more computationally efficient and requires data only from the "best source'. Pre-trained single-source adaptation weights can be directly used for the other datasets, thus offsetting the need for images and GT from all source domains. The improvement in our approach comes from individually distilling relevant information from multiple domains as opposed to considering images from all source domains in every iteration.

### 4.4. Ablation Studies and Additional Experiments

We show the benefits of using multiple sources compared to a single source in Table 1. The multi-source model outperforms the corresponding single source baselines by $3.3\% - 54.15\%$, demonstrating the efficiency of using multiple sources. In boundless DA, we replace the thresholding operator with the KL divergence loss to measure the similarity between the open-set classes and physically similar categories in Table 2. We demonstrate the iteration wise performance of the self-training algorithm and a study of tuning the hyperparameters $\lambda_{distil}$ and $\lambda_{unsup}$ in the supplementary material. Additionally, selecting the best source at the pixel-level degrades accuracy by $18.05$ % due to loss of contextual information. Finally, thresholding on pseudo labels [4] to reduce the number of false positives reduces the mIoU by $4.54$ %.

## 5. Conclusion, limitations and future work

We present a novel learning methods for visual scene understanding in unstructured traffic environment. Our approach consists of a semantic segmentation technique that solves three key aspects of domain adaptation: unsupervised, multi-source and boundless, in unconstrained environments. We present a novel training routine that builds on the ideas of self-training and pseudo-labeling. The self-training routine is used to selectively distil information from various sources by iterative self-training, in addition to exploiting a chosen best source via domain adaptation. In addition, BoMuDANet can identify unknown objects encountered during the testing phase via a simple pseudo labeling strategy. We highlight the benefits of our approach in terms of performing accurate segmentation and visual scene understanding in challenging datasets such as IDD. We highlight improved accuracy over prior methods and perform qualitative evaluation based on a user study.

Our approach has some limitations. Our current approach can only recognize new objects by taking advantage of the structural similarities between various objects in road environments. Currently, our model is unable to detect classes like animals and other classes that do not share any similarities with the 'known' classes. In addition, the existence of multiple 'unknown' objects that share similarities with the same set of 'known' classes can cause inter-class confusion. As part of future work, it would be useful to use the pseudo labeling strategy as a prior, and develop a training method that exploits zero-shot learning strategies. The core step in the self-training algorithm is the selection of a 'best source', which can vary from image to image. Our current formulation does not account for this factor during training. In addition, we do not account for variations within the target dataset, where images can have varying levels of similarities with the source datasets. Future work in this area can focus on an importance weighting scheme for a multi-source domain adaptation network that is more robust. We would also like to evaluate our approach in other challenging scenarios and integrate with planning and navigation.

## References


[1] Bhakti Baheti, Shubham Innani, Suhas Gajre, and Sanjay Talbar. Eff-unet: A novel architecture for semantic segmentation in unstructured environment. In *CVPR*, 2020. 2

[2] Yilei Sun Brenda Goh. Tesla 'very close' to level 5 autonomous driving technology. link, 2020. 1

[3] Maxime Bucher, VU Tuan-Hung, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. In *NIPS*, 2019. 5, 7, 8

[4] Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Buda: Boundless unsupervised domain adaptation in semantic segmentation. *arXiv*, 2020. 2, 3, 7, 8

[5] Mark Campbell, Magnus Egerstedt, Jonathan P How, and Richard M Murray. Autonomous driving in urban environments: approaches, lessons and challenges. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 368(1928), 2010. 1

[6] Rohan Chandra, Uttaran Bhattacharya, Aniket Bera, and Dinesh Manocha. Traphic: Trajectory prediction in dense and heterogeneous traffic using weighted interactions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8483–8492, 2019. 2

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *PAMI*, 40(4), 2017. 1, 2, 4

[8] Yuhua Chen, Wen Li, , and Luc Van Gool. Road: Reality oriented adaptation for semantic segmentation of urban scenes. In *CVPR*, 2018. 8

[9] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 1, 2, 5

[10] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019. 2

[11] Han Guo, Ramakanth Pasunuru, and Mohit Bansal. Multi-source domain adaptation for text classification via distancenet-bandits. In *AAAI*, 2020. 3

[12] Zhiyang Guo, Yingping Huang, Xing Hu, Hongjian Wei, and Baigan Zhao. A survey on deep learning based approaches for scene understanding in autonomous driving. *Electronics*, 10(4):471, 2021. 2

[13] Shijie Hao, Yuan Zhou, and Yanrong Guo. A brief survey on semantic segmentation with deep learning. *Neurocomputing*, 406:302–321, 2020. 2

[14] Markus Hofmarcher, Thomas Unterthiner, José Arjona-Medina, Günter Klambauer, Sepp Hochreiter, and Bernhard Nessler. Visual scene understanding for autonomous driving using semantic segmentation. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pages 285–296. Springer, 2019. 1

[15] Maximilian Jaritz, Tuan-Hung Vu, Raoul de Charette, Emilie Wirbel, and Patrick Pérez. xmuda: Cross-modal unsupervised domain adaptation for 3d semantic segmentation. In *CVPR*, 2020. 2

[16] Tarun Kalluri, Girish Varma, Manmohan Chandraker, and CV Jawahar. Universal semi-supervised semantic segmentation. In *ICCV*, 2019. 8

[17] Tarun Kalluri, Girish Varma, Manmohan Chandraker, and CV Jawahar. Universal semi-supervised semantic segmentation. In *ICCV*, 2019. 7

[18] Taekyung Kim and Changick Kim. Attract, perturb, and explore: Learning a feature alignment network for semi-supervised domain adaptation. *arXiv*, 2020. 2

[19] Divya Kothandaraman, Athira Nambiar, and Anurag Mittal. Domain adaptive knowledge distillation for driving scene semantic segmentation. In *Proceedings of the IEEE/CVF WACV*, pages 134–143. 2

[20] Da Li and Timothy Hospedales. Online meta-learning for multi-source and semi-supervised domain adaptation. *arXiv*, 2020. 2

[21] Guangrui Li, Guoliang Kang, Wu Liu, Yunchao Wei, and Yi Yang. Content-consistent matching for domain adaptive semantic segmentation. 2020. 8

[22] Chuang Lin, Sicheng Zhao, Lei Meng, and Tat-Seng Chua. Multi-source domain adaptation for visual sentiment classification. In *AAAI*, 2020. 3

[23] Dongnan Liu, Donghao Zhang, Yang Song, Fan Zhang, Lauren O'Donnell, Heng Huang, Mei Chen, and Weidong Cai. Unsupervised instance segmentation in microscopy images via panoptic domain adaptation and task re-weighting. In *CVPR*, 2020. 2

[24] Hong Liu, Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Qiang Yang. Separate to adapt: Open set domain adaptation via progressive separation. In *CVPR*, 2019. 3

[25] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. Structured knowledge distillation for semantic segmentation. In *CVPR*, 2019. 5

[26] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 5

[27] Todd K Moon. The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6), 1996. 4

[28] Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *CVPR*, 2020. 2, 8

[29] Pau Panareda Busto and Juergen Gall. Open set domain adaptation. In *ICCV*, 2017. 3

[30] Can Qin, Lichen Wang, Qianqian Ma, Yu Yin, Huan Wang, and Yun Fu. Opposite structure learning for semi-supervised domain adaptation. *arXiv*, 2020. 2

[31] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *ECCV*. Springer, 2016. 2, 5

[32] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *ICCV*, 2019. 2

[33] Kuniaki Saito, Shohei Yamamoto, Yoshitaka Ushiku, and Tatsuya Harada. Open set domain adaptation by backpropagation. In *ECCV*, 2018. 3

[34] Wilko Schwarting, Javier Alonso-Mora, and Daniela Rus. Planning and decision-making for autonomous vehicles. *Annual Review of Control, Robotics, and Autonomous Systems*, 2018. 1

[35] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[36] Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5229–5238, 2019. 2

[37] Marco Toldo, Andrea Maracani, Umberto Michieli, and Pietro Zanuttigh. Unsupervised domain adaptation in semantic segmentation: a review. *arXiv preprint arXiv:2005.10876*, 2020. 3

[38] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *CVPR*, 2018. 2, 4, 5, 6, 8

[39] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017. 2

[40] Girish Varma, Anbumani Subramanian, Anoop Namboodiri, Manmohan Chandraker, and CV Jawahar. Idd: A dataset for exploring problems of autonomous navigation in unconstrained environments. In *WACV*. IEEE, 2019. 1, 5

[41] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*, 2019. 2, 6, 8

[42] Hang Wang, Minghao Xu, Bingbing Ni, and Wenjun Zhang. Learning to combine: Knowledge aggregation for multi-source domain adaptation. *arXiv preprint arXiv:2007.08801*, 2020. 3

[43] Zhonghao Wang, Mo Yu, Yunchao Wei, Rogerio Feris, Jinjun Xiong, Wen-mei Hwu, Thomas S Huang, and Honghui Shi. Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation. In *CVPR*, 2020. 2

[44] Magnus Wrenninge, , , , and Jonas Unger. Synscapes: A photorealistic synthetic dataset for street scene parsing. *arXiv*, 2018. 2, 5

[45] Jianxiong Xiao, Bryan C Russell, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Basic level scene understanding: From labels to structure and beyond. In *SIGGRAPH Asia 2012 Technical Briefs*, pages 1–4. 2012. 1

[46] Jihan Yang, Ruijia Xu, Ruiyu Li, Xiaojuan Qi, Xiaoyong Shen, Guanbin Li, and Liang Lin. An adversarial perturbation oriented domain adaptation approach for semantic segmentation. In *AAAI*, 2020. 2

[47] Luyu Yang, Yogesh Balaji, Ser-Nam Lim, and Abhinav Shrivastava. Curriculum manager for source selection in multi-source domain adaptation. *arXiv preprint arXiv:2007.01261*, 2020. 3

[48] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, 2020. 1, 2, 5

[49] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In *CVPR*, 2017. 1, 2, 4

[50] Yixin Zhang and Zilei Wang. Joint adversarial learning for domain adaptation in semantic segmentation. In *AAAI*, 2020. 2

[51] Han Zhao, Shanghang Zhang, Guanhang Wu, José MF Moura, Joao P Costeira, and Geoffrey J Gordon. Adversarial multiple source domain adaptation. In *NIPS*, 2018. 3, 8

[52] Sicheng Zhao, Bo Li, Pengfei Xu, and Kurt Keutzer. Multi-source domain adaptation in the deep learning era: A systematic survey. *arXiv preprint arXiv:2002.12169*, 2020. 4, 5

[53] Sicheng Zhao, Bo Li, Xiangyu Yue, Yang Gu, Pengfei Xu, Runbo Hu, Hua Chai, and Kurt Keutzer. Multi-source domain adaptation for semantic segmentation. In *NIPS*, 2019. 2, 3, 8

[54] Sicheng Zhao, Guangzhi Wang, Shanghang Zhang, Yang Gu, Yaxian Li, Zhichao Song, Pengfei Xu, Runbo Hu, Hua Chai, and Kurt Keutzer. Multi-source distilling domain adaptation. *arXiv preprint arXiv:1911.11554*, 2019. 3

[55] Zhedong Zheng and Yi Yang. Unsupervised scene adaptation with memory regularization in vivo. *arXiv*, 2019. 2

[56] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 8