Learning Semantic Segmentation from Multiple Datasets with Label Shifts

Dongwan Kim[1], Yi-Hsuan Tsai[3], Yumin Suh[4], Masoud Faraki[4], Sparsh Garg[4], Manmohan Chandraker[4,5], and Bohyung Han[1,2]

[1]ECE & [2]IPAI, Seoul National University
[3]Phiar Technologies [4]NEC Labs America [5]UC San Diego

Abstract. While it is desirable to train segmentation models on an aggregation of multiple datasets, a major challenge is that the label space of each dataset may be in conflict with one another. To tackle this challenge, we propose UniSeg, an effective and model-agnostic approach to automatically train segmentation models across multiple datasets with heterogeneous label spaces, without requiring any manual relabeling efforts. Specifically, we introduce two new ideas that account for conflicting and co-occurring labels to achieve better generalization performance in unseen domains. First, we identify a gradient conflict in training incurred by mismatched label spaces and propose a class-independent binary cross-entropy loss to alleviate such label conflicts. Second, we propose a loss function that considers class-relationships across datasets for a better multi-dataset training scheme. Extensive quantitative and qualitative analyses on road-scene datasets show that UniSeg improves over multi-dataset baselines, especially on unseen datasets, e.g., achieving more than 8%p gain in IoU on KITTI. Furthermore, UniSeg achieves 39.4% IoU on the WildDash2 public benchmark, making it one of the strongest submissions in the zero-shot setting. Our project page is available at https://www.nec-labs.com/~mas/UniSeg.

Keywords: semantic segmentation, multi-dataset training, label shift

1 Introduction

Many segmentation datasets, such as Cityscapes [8], BDD [54], IDD [47] and Mapillary [35], have been leveraged by semantic segmentation models to produce high-quality results [29,28,43,5,55,58,57,60]. However, most approaches only exploit labels within a single dataset for training. Given the expense of labeling segmentation data, it is important to consider whether labels from multiple datasets can be combined to train more robust models.

One benefit of a model trained on multiple datasets would be the increase in data volume and diversity, which allows a joint model to better reason about challenging objects or scenes. Moreover, if a segmentation model could be trained on a label space that is unified across datasets, we may obtain richer training constraints and inference outputs compared to a model trained from a single

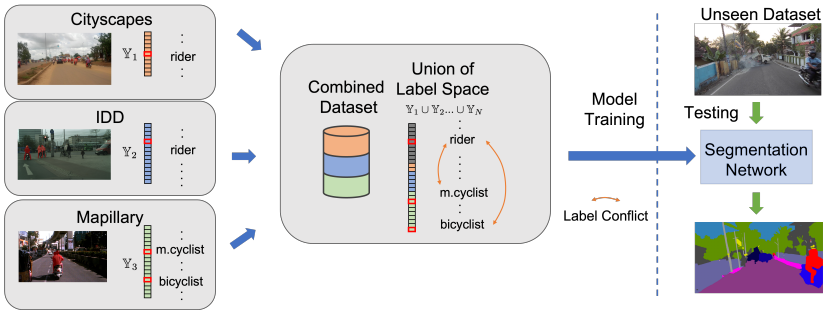

Fig. 1. We tackle the problem of multi-dataset semantic segmentation, where each dataset has a different label space. Directly combining all the datasets and training a model would result in label conflicts within the unified label space. For example, the rider in the right image can be considered as both the "rider" or the "motorcyclist" categories in the unified label space. Therefore, it is important to handle such label conflicts during the training process.

dataset. However, combining multiple datasets is non-trivial since their label spaces are heterogeneous and may possibly be in conflict with one another. For example, Cityscapes has 19 categories while Mapillary has 65 more fine-grained categories. A recent work, MSeg [21], deals with this issue by manually defining a taxonomy for the unified label space, which requires re-annotating many images for consistency across datasets. Such a time-consuming solution limits the scalability to more datasets to be collected in the future.

This paper presents a model-agnostic framework called UniSeg, which allows us to employ multiple datasets with heterogeneous label spaces for training semantic segmentation models. First, we observe that the widely-used cross-entropy (CE) loss often leads to conflicting gradients when two datasets contain categories that are in direct conflict with each other (see Fig. 1), e.g., Cityscapes only has the "rider" class, but Mapillary contains both the "motorcyclist" and "bicyclist" categories. To this end, we consider the binary cross-entropy (BCE) loss for semantic segmentation, which allows us to compute separate gradients for each class and resolve the gradient conflict issue during optimization by selectively ignoring certain classes during loss computation. Surprisingly, this simple modification in the loss term, which we call the *Null BCE*, leads to several significant benefits for multi-dataset training, especially on unseen datasets. Second, to utilize class relations across label spaces, we propose *class-relational BCE*, which allows each pixel to be supervised with multiple labels. For example, if we are able to link "bicyclist" from Mapillary to the Cityscapes "rider" class, the training process can be improved by leveraging such a relationship. Without external knowledge, we infer class relationships and generate multi-class labels that properly link categories across datasets, which are then integrated into the training process with our class-relational BCE loss.

We train segmentation models on the combination of four road-scene datasets for semantic segmentation (Cityscapes [8], BDD [54], IDD [47], and Mapillary [35]), which contains many label conflicts in the unified label space. We then test on three datasets, KITTI [15], CamVid [4] and WildDash [56], which are not used for training and serve as unseen datasets to verify that our model can perform well in other challenging conditions. Our extensive experiments demonstrate the effectiveness of our *UniSeg* framework with the proposed loss terms: Null BCE loss and class-relational BCE loss. We compare against baselines using the traditional multi-dataset training. For instance, on the KITTI dataset, our HRNet-W48 [44] model achieves more than 8%p gain in IoU on average across all the settings, and outperforms other methods on the WildDash2 public benchmark. We also conduct qualitative analyses on the output of our UniSeg model and observe that it makes accurate multi-label predictions, especially for classes with label conflict. In summary, the main contributions of this paper are:

- We design a principled model-agnostic framework for multi-dataset semantic segmentation with label shifts, which is free from additional manual annotation costs and prior knowledge.
- We propose a simple yet effective loss terms to handle the label shift problem, while also introducing a new training scheme via class-relational BCE.
- We validate the benefit of using our method in various multi-dataset training settings, showing significant performance improvements over baselines, especially on unseen datasets during training.

2 Related Work

Multi-dataset Semantic Segmentation. Several recent works have considered to use multiple datasets to jointly train a semantic segmentation model [21,18,27,32,3,50]. However, the adopted setting/goal and the perspective of their approaches vary significantly. For instance, [3] uses both the Mapillary [35] and Cityscapes [8] datasets to train a segmentation model on the Cityscapes label space, while detecting outlier regions in an unseen dataset, WildDash [56]. Moreover, [50] considers multi-dataset training using dataset-specific classifiers on the original label space of each dataset. In contrast to our setting, these methods do not consider using a unified label space with a single classifier.

To exploit different label spaces across datasets, two approaches [27,32] adopt the idea of label hierarchy to jointly train on multiple datasets. However, such a strategy requires a manually pre-defined structure of label space, where categories need to be merged, added, or split in the hierarchy tree. Thus it may not be easily scalable to newly introduced datasets. More recently, MSeg [21] proposes to unify multiple datasets via defining a label taxonomy, which maps all the datasets into the same label space. However, this pre-processing scheme requires human re-annotation on a large number of images to ensure label consistency, which is time-consuming and not scalable to more datasets. In contrast, we aim to tackle the multi-dataset setting with label shifts purely from a model-training perspective without requiring any human intervention.

Multi-label Learning. A few works [12,20,7,59,22] proposed methods for effectively training on multi-label settings, where each data sample is accompanied with one or more labels. As an example, Durand *et al.* [12] propose a partial-BCE loss, which computes the loss only on the known classes while ignoring unknown classes. While there exists a technical similarity to the partial-BCE loss, our idea is derived from a different problem setting than partial labels, i.e., the gradient-conflict in multi-dataset training.To alleviate gradient conflict, our Null-BCE reformulates the problem into a partial label problem, and thus, enables better training without sacrificing label granularity of certain datasets.

Domain Adaptation and Generalization. Unsupervised domain adaptation (DA) techniques have been developed to learn domain-invariant features that reduce the gap across the source and target domains in several tasks, such as image classification [31,14,46,39,23], object detection [6,40,19,17], and semantic segmentation [16,45,48,26,36]. Extending from a traditional dual-domains setting, other DA settings have been proposed to consider multi-source [37] and multi-target [30,9] domains. A more challenging setting, universal DA [38,53], deals with various cases across datasets that may have different label spaces. In our multi-dataset setting with supervisions, although there are also domain gaps across datasets similar to the domain adaptation setting, we focus on solving the label shift and conflict issues, which is orthogonal to the adaptation scenario.

Domain generalization assumes multiple training datasets available, and the goal is to learn a model that can generalize well to unseen datasets. Several methods have been developed via learning a share embedding space [34,11,33,13], domain-specific information [41,24], or meta-learning based approaches [1,25]. However, these approaches mainly focus on the image classification task, and more importantly, assume a shared label space across the training datasets and any unseen ones, which is different from the setting of this paper, where each dataset may have its own distinct label space.

3 Proposed Method

3.1 Multi-dataset Semantic Segmentation

The typical way to optimize a single-dataset semantic segmentation model is to use a pixel-wise cross-entropy loss. When it comes to multiple datasets, since each dataset has its own label space, there could be two straightforward options to train the model. One method is to construct individually separate classifiers for each dataset. However, this could result in problems during testing, where it may not be clear which classifier should be selected. The second option is to unify all the label spaces and train a single classifier, which is more suitable for our problem context and will be the strategy on which this paper focuses.

Cross-Entropy Formulation. Given an image $X_i \in \mathbb{R}^{H \times W \times 3}$ in dataset D_i and its K_i-categorical one-hot label $Y_i \in \{0,1\}^{H \times W \times K_i}$ in the label space $\mathbb{Y}_i$,

we unify the label space as $\mathbb{Y}_u = \mathbb{Y}_1 \cup \mathbb{Y}_2 ... \cup \mathbb{Y}_N$, where N is the number of datasets. Therefore, the original label Y_i is extended to K_u categories, where $K_u \leq \sum_i K_i$ is the number of unified categories. Without any prior knowledge, a natural way to extend Y_i to a K_u-categorical label is to assign all categories in $\mathbb{Y} = \mathbb{Y}_u \setminus \mathbb{Y}_i$ with label 0. As a result, the cross-entropy loss that optimizes the segmentation network $\mathbf{G}$ on multiple datasets can be written as:

$$\mathcal{L}_{seg}^{ce} = -\sum_{i=1}^{N} \sum_{k=1}^{K_u} \sum_{h,w} Y_i^{(h,w,k)} \log(P_i^{(h,w,k)}) \,, \tag{1}$$

where $P_i \in [0,1]^{H \times W \times K_u}$ is the softmax of the segmentation output $O_i = \mathbf{G}(X_i) \in \mathbb{R}^{H \times W \times K_u}$, from the unified classifier. In (1), we omit summation over all samples in each dataset to prevent notations from being over-complicated.

Gradient Conflict in (1). Although unifying the label spaces across datasets enables the standard cross-entropy optimization in (1), it can cause training difficulty when there is a label conflict across datasets. Here, we assume that we do not have prior knowledge regarding the label space and its semantics in the individual datasets. Therefore, such label conflict is likely to occur, as each dataset may define label spaces differently. For instance, Cityscapes only has the "rider" class, while Mapillary does not and has the "motorcyclist" and "bicyclist" categories instead. In this case, the unified label space of $\mathbb{Y}_u$ contains all three categories, "rider", "motorcyclist", and "bicyclist", but during training, images from Mapillary would always treat "rider" with a label of 0.

Based on the example above, such label conflict may cause optimization difficulty with the cross-entropy (CE) loss. To further analyze the negative effect caused by label conflict, we consider the update step of a single parameter θ that contributes to the output O of an arbitrary class k in the last layer of the network. Given an image X_1 from one dataset labeled as k at a position (h, w), the gradient of the loss at a position (h, w) to a parameter θ is calculated as:

$$\frac{\partial \mathcal{L}_{seg}^{ce}}{\partial \theta} = \frac{\partial O_1^{(h,w,k)}}{\partial \theta} (P_1^{(h,w,k)} - Y_1^{(h,w,k)}). \tag{2}$$

Now, consider an identical image X_2 that originates from another dataset with a different label space. Combining the two cases, the gradient for θ becomes:

$$\frac{\partial \mathcal{L}_{seg}^{ce}}{\partial \theta} = \frac{\partial O_1^{(h,w,k)}}{\partial \theta} (P_1^{(h,w,k)} - Y_1^{(h,w,k)}) + \frac{\partial O_2^{(h,w,k)}}{\partial \theta} (P_2^{(h,w,k)} - Y_2^{(h,w,k)}). \tag{3}$$

Note that, since X_1 and X_2 are identical images, $\frac{\partial O_1^{(h,w,k)}}{\partial \theta} = \frac{\partial O_2^{(h,w,k)}}{\partial \theta}$. However, since the two images originate from different datasets, we have $Y_1^{(h,w,k)} \neq Y_2^{(h,w,k)}$ (*i.e.*, if $Y_1^{(h,w,k)} = 1$, $Y_2^{(h,w,k)} = 0$). Thus, the parameter θ receives one gradient that is smaller than 0, and another that is larger than 0, despite coming from identical samples. This is not optimal for training the model, yet can easily occur when training a model on multiple datasets with conflicting label spaces.

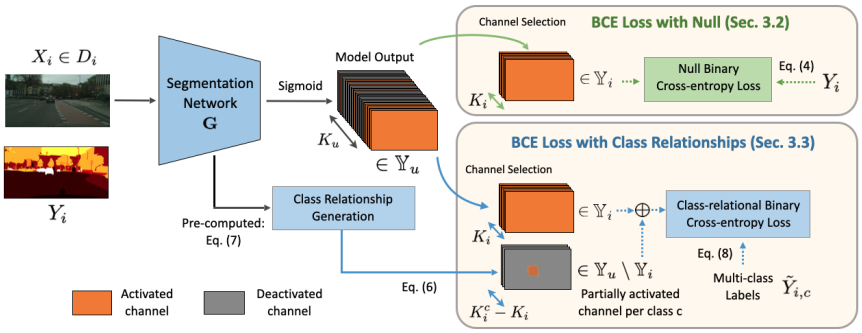

Fig. 2. Overview of the proposed framework using the Null BCE loss (Section 3.2) and Class-relational BCE loss (Section 3.3): 1) For Null BCE, we replace the original CE loss function to reduce the gradient conflict issue as mentioned in Section 3.1. In the loss calculation, we only take the categories in the label space $\mathbb{Y}_i$ of D_i into consideration via (4); 2) For class-relational BCE, through the pre-computed class relationships for each dataset via (8), we incorporate the generated multi-class labels $\tilde{Y}_{i,c}$ via (6) to form (9). Note that only some pixels within the image may have multi-class labels, as illustrated (highlighted in orange).

3.2 Revisited Binary Cross-Entropy Loss

To resolve the aforementioned issue, we find that the binary cross-entropy (BCE) loss, while similar to the CE loss, exhibits some interesting properties that are desirable for our task setting. First, it does not require a softmax operation, whose value is dependent on the output logits of other classes. Instead, BCE loss is accompanied by a point-wise sigmoid activation. Furthermore, with the BCE loss, we are able to selectively assign labels to each class. Therefore, we design a "Null" class strategy, where we only assign the *valid* labels for each dataset. That is, for images from the D_i dataset, we only assign labels for categories within $\mathbb{Y}_i$, while for other categories $\mathbb{Y} = \mathbb{Y}_u \setminus \mathbb{Y}_i$, we neither assign label 0 nor 1. We name this loss as the "Null BCE loss", which is written as:

$$\mathcal{L}_{seg}^{bce} = -\sum_{i=1}^{N}\sum_{k=1}^{K_i}\sum_{h,w} Y_i^{(h,w,k)} \log(Q_i^{(h,w,k)}) + (1 - Y_i^{(h,w,k)}) \log(1 - Q_i^{(h,w,k)}) \,, \tag{4}$$

where $Q_i \in [0,1]^{H \times W \times K_u}$ is the output from the sigmoid activation. It is important to note that, although there is only a slight difference from (1) in the summation of the loss term, *i.e.*, summed over K_i instead of K_u, this change resolves the gradient conflict issue mentioned in (3) since no loss is calculated for class k for the input image X_2 (see the example in (3)):

$$\frac{\partial \mathcal{L}_{seg}^{bce}}{\partial \theta} = \frac{\partial O_1^{(h,w,k)}}{\partial \theta}(Q_1^{(h,w,k)} - Y_1^{(h,w,k)}). \tag{5}$$

The procedure is illustrated in the top-right of Fig. 2. A more detailed derivation for both (2) and (5) is provided in the supplementary material.

3.3 Class-relational Binary Cross-Entropy Loss

Another advantage of BCE over the CE loss is that it can be used to train a model with multi-label supervision. While our Null BCE loss in Section 3.2 alleviates the gradient conflict issue caused by inconsistent label spaces across multiple datasets, it simply chooses to ignore classes that are not within the label space of a given sample. Thus, we propose another loss that better utilizes the inter-class relationships by explicitly providing multi-label supervision at pixels where co-occurring labels may exist (bottom-right of Fig. 2).

Multi-class Label Generation. For a class c from dataset D_i, we generate the new multi-class label $\tilde{Y}_{i,c} \in \{0,1\}^{K_u}$. This aims at training the classifier to predict not only the original class but also any co-existing class(es) from the unified label space $\mathbb{Y}_u$. For example, multi-class labels can be generated for subset/superset relationships, *e.g.*, "bicyclist" with "rider" or "crosswalk" with "road", but not classes with similar appearance or high co-occurrence.

Assuming we know these class relationships, we assign additional label $c' \in \mathbb{Y}_u$ only if its similarity to the class $c \in \mathbb{Y}_i$ is sufficiently large,

$$\tilde{Y}_{i,c}^{(c')} = \begin{cases} 1 & \text{if } c' = c \text{ or } \mathbf{s}_{i,c}^{(c')} > \max(\tau, \mathbf{s}_{i,c}^{(c)}) \\ \emptyset & \text{else if } c' \in \mathbb{Y}_u \setminus \mathbb{Y}_i \\ 0 & \text{otherwise} \end{cases}, \tag{6}$$

where $\mathbf{s}_{i,c}^{(c')}$ is the similarity between class c and c' measured in dataset D_i (details for calculation are introduced in the next section) and τ is a threshold. When classes c and c' have a conflict, *e.g.*, when c is "bicyclist" and c' is "rider", we expect the similarity $\mathbf{s}_{i,c}^{(c')}$ to be large. In contrast, we expect it to be small for classes without conflict.

For the choice of τ, we check if the class of the largest score in $\mathbf{s}_{i,c}$ comes from another dataset, D_j, which indicates a high chance of label conflict, and thus, requires multi-class labels. For such cases, we average the largest scores and obtain a value of 0.48 ± 0.01, which is used as the threshold τ. Note that, the max condition in (6) implies that multi-labels are only activated, *i.e.*, $\tilde{Y}_{i,c}^{(c')} = 1$, when similarity for class $c' \in \mathbb{Y}_u \setminus \mathbb{Y}_i$ is higher than that of the original class c, *i.e.*, $\mathbf{s}_{i,c}^{(c')} \geq \mathbf{s}_{i,c}^{(c)}$. This makes label generation more robust to variations in τ. Fig. 3 illustrates an example of this process.

Class Relationship Generation. To extract inter-class relationships, we leverage the cosine classifier [49], such that the cosine similarity between the feature and any classifier weight vector can be calculated, even for classes across datasets. Let $\hat{\phi}_c$ denote the ℓ_2-normalized 1×1 convolution weight vector for the c^{th} class,

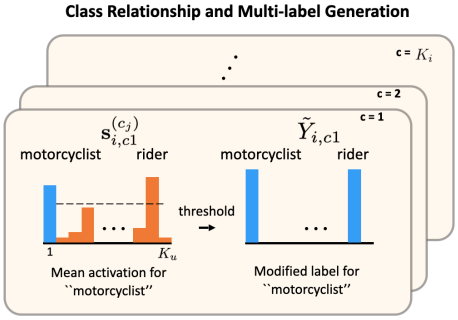

Fig. 3. One example of generating the final multi-class label $\tilde{Y}_{i,c}$ through the mean activation $\mathbf{s}_{i,c}^{(c)}$ for "motorcyclist", where the final multi-class includes the "rider" class.

and $\hat{\mathbf{x}}^{(h,w)}$ denote the ℓ_2-normalized input feature vector at location (h, w). Then, the cosine similarity, $S^{(h,w,c)}$, for class c at location (h, w) is calculated as:

$$S^{(h,w,c)} = t \cdot \hat{\phi}_{\mathbf{c}}^{\top} \hat{\mathbf{x}}^{(h,w)} = t \cdot ||\phi_{\mathbf{c}}|| ||\mathbf{x}^{(h,w)}|| \cos\theta_c, \tag{7}$$

where θ_c represents the angle between ϕ_c and $\mathbf{x}^{(h,w)}$, and t is a scaling factor.

We then calculate the mean activation vector of the final output layer as the similarity score, $\mathbf{s}_{i,c} \in [0, 1]^{K_u}$, which indicates the relationships between each class c in dataset D_i and all other classes in the unified label space $\mathbb{Y}_u$.

$$\mathbf{s}_{i,c}^{(c')} = \frac{1}{M_{i,c}} \sum_{X_i \in D_i} \sum_{h,w} S_i^{(h,w,c')} \cdot \mathbb{1}_{i,c}^{(h,w)}, \quad \forall i \in \{1, ..., N\}; \forall c' \in \mathbb{Y}_u, \tag{8}$$

where $M_{i,c}$ denotes the number of pixels with ground-truth of c in D_i, X_i represents the samples in D_i, and $\mathbb{1}_{i,c}^{(h,w)} \in \{0, 1\}$ is an indicator whose value is 1 if the ground-truth is c at location (h, w) of X_i. Note that, $\mathbf{s}_{i,c}$ can be computed either for each dataset or over all the datasets. In practice, we adopt the dataset-specific similarities to reflect the properties of each individual dataset.

Discussions. In (8), we define the similarity between classes to be asymmetric, *i.e.*, $s_{i,c}^{c'} \neq s_{j,c'}^{c}$, where $i \neq j$ and $c' \in \mathbb{Y}_j$, in order to address the asymmetric relations such as subset/superset. For example, since "rider" is a superset of "motorcyclist", any "motorcyclist" is also a "rider", yet the opposite is not always true. Here, our method is able to implicitly capture such intricate relationships, where the model can generate stronger "rider" activations given inputs of "motorcyclist". On the contrary, the model does not generate strong "motorcyclist" activations on "rider", since a "rider" is not always a "motorcyclist".

Class-relational BCE Loss. With the the multi-class label $\tilde{Y}_{i,c}$ via (6) that is aware of the class-relationships across datasets, we define our class-relational

BCE Loss as:

$$\mathcal{L}_{seg}^{cl-bce} = -\sum_{i=1}^{N}\sum_{k=1}^{K_i^c}\sum_{h,w} \tilde{Y}_{i,c}^{(h,w,k)} \log(Q_i^{(h,w,k)}) + (1-\tilde{Y}_{i,c}^{(h,w,k)}) \log(1-Q_i^{(h,w,k)}) \,, \tag{9}$$

where the difference from (4) is the summation over the K_i^c-categorical multi-label $\tilde{Y}_{i,c}$ that is calculated for each class c. As a result, some of the "Null" categories from Section 3.2 can now be incorporated in the loss calculation based on the inferred class relationships. The full list of generated multi-labels are provided in the supplementary material.

3.4 Model Training and Implementation Details

Data Preparation. As noted in Section 3.1, given N number of datasets, $D = \{D_1, D_2, ..., D_N\}$, we unify the label space as the union of the N individual label spaces, $\mathbb{Y}_u = \mathbb{Y}_1 \cup \mathbb{Y}_2 ... \cup \mathbb{Y}_N$. We concatenate the N datasets to obtain a single unified dataset. Before doing so, we preprocess each dataset such that the segmentation labels can be re-mapped to the correct index of $\mathbb{Y}_u$. Note that, to make the training batches consistent, we resize all the images with the shorter side as 1080 pixels, and use 713×713 random cropping with standard data augmentations such as random scaling and random horizontal flipping.

Implementation Details. We use the HRNet V2 [44] backbones initialized with weights pre-trained on ImageNet [10]. The batch size is 32/16 with an initial learning rate of 0.02/0.01 for HRNet-W18 and HRNet-W48, respectively. All models are trained using SGD with 0.9 momentum and a polynomial learning rate decay scheme for 150 epochs. To obtain the multi-class labels in our class-relational BCE loss, we pre-train an HRNet-W18 model using the cosine classifier and fix the generated class relationships $\mathbf{s}_{i,c}$ for each dataset in all the experiments.

4 Experimental Results

In this section, we first introduce our experimental setting on multi-dataset semantic segmentation. Then, to verify the robustness of our UniSeg model, we present the results trained using different combinations of four driving-scene datasets (Cityscapes [8], BDD [54], IDD [47], and Mapillary [35]), and tested on three unseen datasets (KITTI [15], CamVid [4] and WildDash [56]). Note that we experiment on road-scenes datasets to better highlight the negative effects of label conflict, and to demonstrate that UniSeg can effectively alleviate the label conflict issue. In addition, we diversify our experiments by training models on "Leave-One-Out" settings, where one of the four training datasets acts as a held-out testing set (unseen), and the model is trained on the remaining three datasets. Our quantitative results are accompanied by qualitative results, which provide more insight of our model.

Table 1. mIoU comparisons with baselines using the HRNet-W18 architecture. KITTI, WildDash, and CamVid are fixed as unseen test datasets. "N/A" indicates the setting where there are no multi-labels generated for our final model and thus we only show our Null BCE setting. "C-R BCE" indicates class-relational BCE.

Train Datasets	Method	KITTI	WildDash	CamVid	Mean
Single-dataset	Single-best (CE)	41.9	41.2	60.8	48.0
C + I + B	Multi-dataset (CE)	46.2	44.3	70.3	51.6
	UniSeg: Null | C-R BCE	**54.9** | N/A	**44.6** | N/A	**71.8** | N/A	**54.9** | N/A
I + B + M	Multi-dataset (CE)	48.0	47.5	71.2	55.4
	UniSeg: Null | C-R BCE	52.7 | **54.6**	47.7 | **48.3**	**73.2** | 72.7	57.9 | **58.5**
C + I + M	Multi-dataset (CE)	50.2	41.3	72.8	54.8
	UniSeg: Null | C-R BCE	55.9 | **56.5**	44.1 | **44.8**	73.3 | **73.8**	57.8 | **58.4**
C + B + M	Multi-dataset (CE)	55.0	46.2	73.4	58.2
	UniSeg: Null | C-R BCE	59.0 | **59.2**	47.5 | **48.7**	73.8 | **74.0**	60.1 | **60.6**
C + I + B + M (All)	Multi-dataset (CE)	48.8	46.0	72.7	55.8
	UniSeg: Null | C-R BCE	57.6 | **58.9**	47.5 | **48.2**	73.3 | **73.9**	59.5 | **60.3**

4.1 Datasets and Experimental Setting

Here, we describe individual datasets and their dataset-specific characteristics that could affect multi-dataset training on semantic segmentation. In the experiments, we use the official splits for training and evaluation.

The **Cityscapes** and **BDD** datasets are collected from different environments (central Europe and USA, respectively), but both contain the same 19 classes in their label spaces. The **IDD** dataset is collected in India, and provides a hierarchical label space with four levels. We follow a conventional level-3 setting which contains 26 classes. Finally, the **Mapillary** dataset is one of the largest driving scenes dataset, with data collected from around the world, and has a total of 65 fine-grained categories. Overall, the unified label space when training on all four datasets has a total of 70 categories.

KITTI, **WildDash**, and **CamVid** are all small-scale datasets that we use as unseen test datasets. The label spaces of KITTI and WildDash are identical to Cityscapes, while for CamVid, we follow the reduced label space used in [21].

Evaluation. For quantitative evaluation, we follow the standard evaluation protocol for the single class prediction, for simplicity. Specifically, we evaluate on the classes that exist both on the label set where the model is trained from and the label set defined in the test dataset. We select appropriate channels from the model output followed by the argmax operation. Note that our method can also predict the co-occurring categories for each pixel, as demonstrated qualitatively in Section 4.4. During testing, following [21], all images are resized so that the height of the image is 1080p (while maintaining the aspect ratio). Intersection-over-union (IoU) score is used to evaluate the segmentation output.

Table 2. mIoU comparisons with baselines using the HRNet-W48 architecture.

Train Datasets	Method	KITTI	WildDash	CamVid	Mean
Single-dataset	Single-best (CE)	48.1	48.8	73.6	56.8
C + I + B	Multi-dataset (CE)	54.5	51.5	76.7	60.9
	UniSeg: Null | C-R BCE	**62.9** | N/A	**54.2** | N/A	**76.8** | N/A	**64.6** | N/A
I + B + M	Multi-dataset (CE)	54.4	53.5	77.7	61.9
	UniSeg: Null | C-R BCE	58.4 | **59.3**	**55.9** | 55.7	77.8 | **78.3**	64.0 | **64.4**
C + I + M	Multi-dataset (CE)	56.8	52.9	78.0	62.6
	UniSeg: Null | C-R BCE	63.9 | **65.8**	53.5 | **53.6**	78.4 | **78.7**	65.3 | **66.0**
C + B + M	Multi-dataset (CE)	57.2	54.2	78.2	63.2
	UniSeg: Null | C-R BCE	64.7 | **68.0**	55.3 | **57.8**	78.1 | **78.3**	66.0 | **68.0**
C + I + B + M (All)	Multi-dataset (CE)	57.0	56.0	78.1	63.7
	UniSeg: Null | C-R BCE	64.4 | **65.2**	56.5 | **58.4**	78.2 | **78.6**	66.4 | **67.4**

Table 3. mIoU comparisons with baselines using SegFormer-B1 and B4 architectures.

Method	Arch.	KITTI	WildDash	CamVid	Mean
Multi-dataset (CE)	SegFormer B1	55.1	51.4	75.0	60.5
UniSeg: Null | C-R BCE		59.7 | **60.3**	53.4 | **54.3**	75.5 | **75.7**	62.9 | **63.4**
Multi-dataset (CE)	SegFormer B4	58.4	57.0	78.4	64.6
UniSeg: Null | C-R BCE		**69.1** | 69.0	60.1 | **61.9**	79.2 | **79.6**	69.5 | **70.2**

4.2 Overall Performance

We present our quantitative results for the unseen datasets in Tables 1, 2, and 3, and the leave-one-out setting in Table 4. For more insightful comparisons, we focus on the evaluation of unseen datasets as it is a more interesting setting for validating the generalizability of models, and leave the results for seen datasets in the supplementary material.

Full Setting. In Tables 1 and 2, we show the performance on unseen datasets (KITTI, WildDash, CamVid) for various combinations of the training datasets (*i.e.*, combinations of three or four datasets). In addition to the three methods, we present the "single-best" baseline, where the results are obtained by the best model after training on each of the datasets using the CE loss. That is, we evaluate all single-dataset models and report the strongest result for each unseen dataset. In Table 3, we present the performance of SegFormer [51] models (B1 and B4) trained on all four datasets to demonstrate that UniSeg is model-agnostic.

Leave-One-Out. We employ Leave-One-Out settings with the training datasets to diversify our evaluation. In these settings, one of the four training datasets (*i.e.*, Cityscapes, IDD, BDD, Mapillary) is left out of training and treated as an unseen test dataset. For example, in Table 4, each column presents results of

Table 4. mIoU comparisons for the Leave-One-Out settings with HRNet-W18 and HRNet-W48. Each column indicates the unseen testing dataset, while the model is trained on the remaining three datasets.

Method	Arch.	Cityscapes	IDD	BDD	Mapillary	Mean
Multi-dataset (CE)	HRNet W18	55.0	44.8	52.2	45.7	49.4
UniSeg: Null | C-R BCE		56.1 | **56.8**	46.2 | **47.6**	**52.3** | 52.1	**48.1** | **48.1**	50.7 | **51.2**
Multi-dataset (CE)	HRNet W48	62.1	49.2	56.8	51.8	55.0
UniSeg: Null | C-R BCE		64.6 | **65.8**	**53.3** | 52.9	**58.1** | 57.9	**53.4** | **53.4**	57.4 | **57.5**

the unseen test dataset, while the other three serve as train datasets. Note that, when training on "Cityscapes, IDD, BDD", the unified label space is small and no new labels are generated by (6) (*i.e.*, column "Mapillary" of Table 4).

Results. First, we observe that jointly training on multiple datasets generally outperforms the single-best setting, even when the CE loss is used. This shows the advantage of using multi-dataset training, where data diversity and volume are increased. We observe that the two variants of our UniSeg models — Null BCE and class-relational BCE — consistently perform favorably against the multi-dataset baseline with the typical CE loss. For example, using either the HRNet-W18/W-48 model, averaged over all the settings, there is a 7.2%/8.2% gain on KITTI and a 3.3%/3.6% gain on "Mean" in Table 1 and 2, which is considered as a significant improvement in semantic segmentation. Furthermore, the results in Tables 3 (different model architecture) and 4 (diverse dataset combinations) follow a similar trend to Tables 1 and 2, where the UniSeg models consistently outperform the CE baseline. This validates our original intuition that the gradient conflict in the CE loss affects model's robustness, regardless of the model architecture (CNNs in Tables 1 and 2 and Transformers in Table 3) and dataset combination. Finally, comparing between our two model variants, class-relational BCE further improves the overall performance. This shows that providing our generated multi-class labels helps multi-dataset training with conflicting label spaces.

4.3 Results on WildDash2 Benchmark

We further highlight the effectiveness of UniSeg by evaluating on the WildDash2 (WD2) benchmark. The WD2 benchmark is a newer version of the original WildDash dataset, with a few additional classes and negative samples. To evaluate on the WD2 benchmark, we employ the HRNet-W48 "Multi-dataset" and UniSeg models trained on all four datasets (C + I + B + M setting of Table 2). Only the test images are provided to users, while evaluation is done on the WD2 server.

Our UniSeg model currently sits at the fourth place on the public leaderboard[1], only surpassed by three submissions from a single method that uses a more

[1] https://wilddash.cc/benchmark/summary_tbl?hc=semantic_rob_2020

Table 5. Class mIoU and negative class mIoU on the WildDash2 benchmark.

Method	Architecture	Class mIoU	Negative mIoU	Meta Avg.
MSeg [21]	HRNet-W48	38.7	24.7	35.2
Yin *et al.* [52]	HRNet-W48	-	-	35.7
Yin *et al.* [52]	Segformer-B5	-	-	37.9
Multi-Dataset (CE)	HRNet-W48	39.0	27.9	36.0
UniSeg (C-R BCE)	HRNet-W48	**41.7**	**34.8**	**39.4**

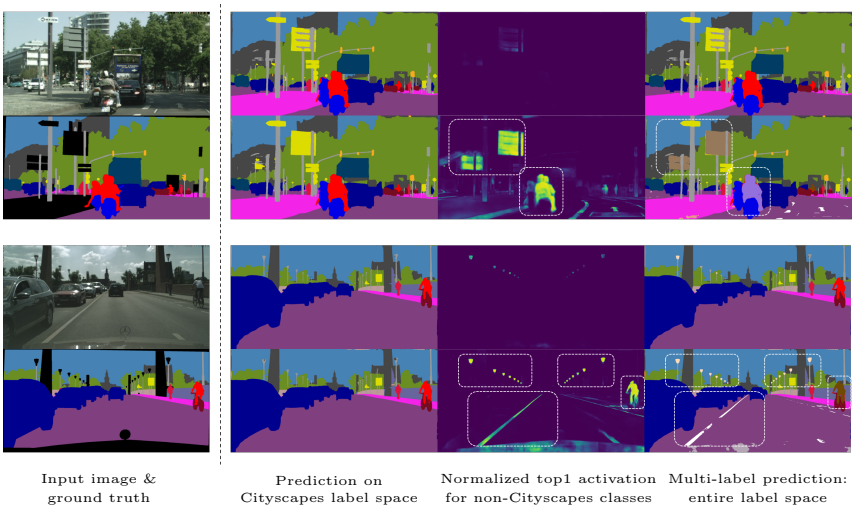

Fig. 4. Multi-label predictions on two samples of Cityscapes. For each set of outputs, the first row corresponds to an HRNet-W48 model trained with the CE loss, while the second row corresponds to our C-R BCE model. While both models make strong predictions on the Cityscapes label space (column 2), only the C-R BCE model has high (normalized) activations for non-Cityscapes classes in regions with label conflict (column 3). For example, the C-R BCE model correctly predicts "traffic sign - back" (light brown) and "traffic light" (beige), even though it is not included in the ground-truth for Cityscapes. Furthermore, the C-R BCE model can make more fine-grained predictions, such as "rider" → "motorcyclist" (light purple), "rider" → "bicyclist" (brown), and "road" → "lane marking" (white).

powerful architecture and includes WD2 in the training set. In contrast, we do not use any WD2 data during training. A summary of the results in shown in Table 5. Note that, while MSeg [21] merges some important fine-grained classes such as "road markings", our UniSeg model is able to make predictions for such classes. This highlights the benefits of retaining the original fine-grained labels. Here, we also compare with Yin *et al.* [52] which facilitates multi-dataset training by replacing class labels with text descriptions, while also using open datasets [2,42] for training.

4.4 Qualitative Analysis

To better understand the full capacity of our UniSeg model, we visualize the output predictions of the UniSeg (C-R BCE) and CE models on two samples of the Cityscapes validation set in Fig. 4. We first normalize the logits for each model's output, which is done by computing the softmax across all 70 classes for the CE model, and an element-wise sigmoid operation for the UniSeg model. The classes of Cityscapes with the top-1 scores are plotted to obtain the predictions in column 2. Next, we identify the top-1 classes among the non-Cityscapes classes and plot the scores in column 3. Finally, we obtain multi-label predictions by thresholding the scores of the non-Cityscapes classes: if the score is above a set threshold, the original class is replaced with the non-Cityscapes class. Here, we use 0.5 as the threshold for the UniSeg model, and 0.1 for the CE model.

Through this visualization, we observe that our model exhibits interesting properties beyond quantitative results. First, we find that our model can **make accurate predictions even for regions where Cityscapes does not provide a ground truth**: in the first sample, although the backside of the traffic signs are not labeled (black in ground truth) in Cityscapes, our model outputs high scores for these pixels (column 3) and overrides the original prediction to the "traffic sign - back" class from the Mapillary dataset (column 4). Furthermore, in the second sample, we observe similar behavior for "street lights" (beige).

Our model also **effectively handles cases with direct label conflict**. In the first sample we see men on a motorcycle, which is given the "rider" label in Cityscapes (column 2). However, since our model is also trained on the "motorcyclist" class, and is able to alleviate the gradient conflict between "rider" and "motorcyclist", our model generates large activations for "motorcyclist" (light purple in column 4) as well. Similar results can be seen for the second sample, where the "lane marking" class (white) replaces parts of the "road" class, and the "bicyclist" class (brown) overrides the "rider" class. Note that, unlike the UniSeg model, the model trained on CE cannot produce large activations for these conflicting labels.

5 Conclusion

In this paper, we proposed UniSeg, which is an effective method to train multi-dataset segmentation models with different label spaces. To alleviate the gradient conflict issue caused by conflicting labels across datasets, we designed a "Null" class strategy using the class-independent BCE loss. To further reap the benefits of multi-dataset training, we incorporated learned class-relationships into the class-relational BCE loss. Our experiments demonstrate that UniSeg improves performance over ordinary multi-dataset training, especially for the unseen datasets.

Acknowledgements. This work was done during the first author's internship at NEC Labs America, and was supported by the NRF Korea grant [No. 2022R1A2C3012210] and the IITP grants [No. 2021-0-01343] funded by the Korean government (MSIT).

References

1. Balaji, Y., Sankaranarayanan, S., Chellappa, R.: Metareg: Towards domain generalization using meta-regularization. In: NeurIPS (2018) 4
2. Benenson, R., Popov, S., Ferrari, V.: Large-scale interactive object segmentation with human annotators. In: CVPR (2019) 13
3. Bevandic, P., Kreso, I., Orsic, M., Segvic, S.: Simultaneous semantic segmentation and outlier detection in presence of domain shift. In: German Conference on Pattern Recognition (2019) 3
4. Brostow, G.J., Fauqueur, J., Cipolla, R.: Semantic object classes in video: A high-definition ground truth database. Pattern Recognit. Lett. **30**(2), 88–97 (2009) 3, 9
5. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. CoRR **abs/1606.00915** (2016) 1
6. Chen, Y., Li, W., Sakaridis, C., Dai, D., Gool, L.V.: Domain adaptive faster r-cnn for object detection in the wild. In: CVPR (2018) 4
7. Cole, E., Mac Aodha, O., Lorieul, T., Perona, P., Morris, D., Jojic, N.: Multi-label learning from single positive labels. In: CVPR (2021) 4
8. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: CVPR (2016) 1, 3, 9
9. Dai, S., Sohn, K., Tsai, Y.H., Carin, L., Chandraker, M.: Adaptation across extreme variations using unlabeled domain bridges. arXiv preprint arXiv:1906.02238 (2019) 4
10. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009) 9
11. Dou, Q., Castro, D.C., Kamnitsas, K., Glocker, B.: Domain generalization via model-agnostic learning of semantic features. In: NeurIPS (2019) 4
12. Durand, T., Mehrasa, N., Mori, G.: Learning a deep convnet for multi-label classification with partial labels. In: CVPR (2019) 4
13. Faraki, M., Yu, X., Tsai, Y.H., Suh, Y., Chandraker, M.: Cross-domain similarity learning for face recognition in unseen domains. In: CVPR (2021) 4
14. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.: Domain-adversarial training of neural networks. In: JMLR (2016) 4
15. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: CVPR (2012) 3, 9
16. Hoffman, J., Tzeng, E., Park, T., Zhu, J.Y., Isola, P., Saenko, K., Efros, A.A., Darrell, T.: Cycada: Cycle-consistent adversarial domain adaptation. In: ICML (2018) 4
17. Hsu, C.C., Tsai, Y.H., Lin, Y.Y., Yang, M.H.: Every pixel matters: Center-aware feature alignment for domain adaptive object detector. In: ECCV (2020) 4
18. Kalluri, T., Varma, G., Chandraker, M., Jawahar, C.V.: Universal semi-supervised semantic segmentation. In: ICCV (2019) 3
19. Kim, T., Jeong, M., Kim, S., Choi, S., Kim, C.: Diversify and match: A domain adaptive representation learning paradigm for object detection. In: CVPR (2019) 4
20. Kundu, K., Tighe, J.: Exploiting weakly supervised visual patterns to learn from partial annotations. In: Advances in Neural Information Processing Systems (2020) 4

21. Lambert, J., Liu, Z., Sener, O., Hays, J., Koltun, V.: Mseg: A composite dataset for multi-domain semantic segmentation. In: CVPR (2020) 2, 3, 10, 13
22. Lanchantin, J., Wang, T., Ordonez, V., Qi, Y.: General multi-label image classification with transformers. In: CVPR (2021) 4
23. Lee, S., Kim, D., Kim, N., Jeong, S.G.: Drop to adapt: Learning discriminative features for unsupervised domain adaptation. In: ICCV (2019) 4
24. Li, D., Yang, Y., Song, Y.Z., Hospedales, T.M.: Deeper, broader and artier domain generalization. In: ICCV (2017) 4
25. Li, D., Yang, Y., Song, Y.Z., Hospedales, T.M.: Learning to generalize: Meta-learning for domain generalization. In: AAAI (2018) 4
26. Li, Y., Yuan, L., Vasconcelos, N.: Bidirectional learning for domain adaptation of semantic segmentation. In: CVPR (2019) 4
27. Liang, X., Zhou, H., Xing, E.: Dynamic-structured semantic propagation network. In: CVPR (2018) 3
28. Lin, G., Shen, C., van dan Hengel, A., Reid, I.: Efficient piecewise training of deep structured models for semantic segmentation. In: CVPR (2016) 1
29. Liu, Z., Li, X., Luo, P., Loy, C.C., Tang, X.: Semantic Image Segmentation via Deep Parsing Network. In: ICCV (2015) 1
30. Liu, Z., Miao, Z., Pan, X., Zhan, X., Lin, D., Yu, S.X., Gong, B.: Open compound domain adaptation. In: CVPR (2020) 4
31. Long, M., Cao, Y., Wang, J., Jordan, M.: Learning transferable features with deep adaptation networks. In: ICML (2015) 4
32. Meletis, P., Dubbelman, G.: Training of convolutional networks on multiple heterogeneous datasets for street scene semantic segmentation. In: IEEE Intelligent Vehicles Symposium (IV) (2018) 3
33. Motiian, S., Piccirilli, M., Adjeroh, D.A., Doretto, G.: Unified deep supervised domain adaptation and generalization. In: ICCV (2017) 4
34. Muandet, K., Balduzzi, D., Schölkopf, B.: Domain generalization via invariant feature representation. In: ICML (2013) 4
35. Neuhold, G., Ollmann, T., Bulo, S.R., Kontschieder, P.: The mapillary vistas dataset for semantic understanding of street scenes. In: ICCV (2017) 1, 3, 9
36. Paul, S., Tsai, Y.H., Schulter, S., Roy-Chowdhury, A.K., Chandraker, M.: Domain adaptive semantic segmentation using weak labels. In: ECCV (2020) 4
37. Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B.: Moment matching for multi-source domain adaptation. In: ICCV (2019) 4
38. Saito, K., Kim, D., Sclaroff, S., Saenko, K.: Universal domain adaptation through self-supervision. In: NeurIPS (2020) 4
39. Saito, K., Watanabe, K., Ushiku, Y., Harada, T.: Maximum classifier discrepancy for unsupervised domain adaptation. In: CVPR (2018) 4
40. Saito1, K., Ushiku, Y., Harada, T., Saenko, K.: Strong-weak distribution alignment for adaptive object detection. In: CVPR (2019) 4
41. Seo, S., Suh, Y., Kim, D., Kim, G., Han, J., Han, B.: Learning to optimize domain specific normalization for domain generalization. In: ECCV (2020) 4
42. Shao, S., Li, Z., Zhang, T., Peng, C., Yu, G., Zhang, X., Li, J., Sun, J.: Objects365: A large-scale, high-quality dataset for object detection. In: ICCV (2019) 13
43. Shelhamer, Evan an Long, J., Darrell, T.: Fully convolutional networks for semantic segmentation. TPAMI (2016) 1
44. Sun, K., Zhao, Y., Jiang, B., Cheng, T., Xiao, B., Liu, D., Mu, Y., Wang, X., Liu, W., Wang, J.: High-resolution representations for labeling pixels and regions. arXiv:1904.04514 (2019) 3, 9

45. Tsai, Y.H., Sohn, K., Schulter, S., Chandraker, M.: Domain adaptation for structured output via discriminative patch representations. In: ICCV (2019) 4
46. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: CVPR (2017) 4
47. Varma, G., Subramanian, A., Namboodiri, A.M., Chandraker, M., Jawahar, C.V.: Idd: A dataset for exploring problems of autonomous navigation in unconstrained environments. In: WACV (2019) 1, 3, 9
48. Vu, T.H., Jain, H., Bucher, M., Cord, M., Pérez, P.: Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In: CVPR (2019) 4
49. Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W.: Cosface: Large margin cosine loss for deep face recognition. In: CVPR (2018) 7
50. Wang, L., Li, D., Zhu, Y., Tian, L., Shan, Y.: Cross-dataset collaborative learning for semantic segmentation. In: CVPR (2021) 3
51. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. In: NeurIPS (2021) 11
52. Yin, W., Liu, Y., Shen, C., van den Hengel, A., Sun, B.: The devil is in the labels: Semantic segmentation from sentences. arXiv:2202.02002 (2022) 13
53. You, K., Long, M., Cao, Z., Wang, J., Jordan, M.I.: Universal domain adaptation. In: CVPR (2019) 4
54. Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., Darrell, T.: Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In: CVPR (2020) 1, 3, 9
55. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. In: ICLR (2016) 1
56. Zendel, O., Honauer, K., Murschitz, M., Steininger, D., Dominguez, G.F.: Wilddash - creating hazard-aware benchmarks. In: ECCV (2018) 3, 9
57. Zhang, H., Dana, K., Shi, J., Zhang, Z., Wang, X., Tyagi, A., Agrawal, A.: Context encoding for semantic segmentation. In: CVPR (2018) 1
58. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR (2017) 1
59. Zhao, X., Schulter, S., Sharma, G., Tsai, Y.H., Chandraker, M., Wu, Y.: Object detection with a unified label space from multiple datasets. In: ECCV (2020) 4
60. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., Torr, P.H.: Conditional random fields as recurrent neural networks. In: ICCV (2015) 1