Cross-Dataset Collaborative Learning for Semantic Segmentation

Li Wang [1], Dong Li [1], Yousong Zhu[2], Lu Tian [1], Yi Shan [1]
[1] Xilinx Inc., Beijing, China.
[2] Institute of Automation, Chinese Academy of Sciences, Beijing, China.
{liwa, dongl, lutian, yishan}@xilinx.com, yousong.zhu@nlpr.ia.ac.cn

Abstract

Recent work attempts to improve semantic segmentation performance by exploring well-designed architectures on a target dataset. However, it remains challenging to build a unified system that simultaneously learns from various datasets due to the inherent distribution shift across different datasets. In this paper, we present a simple, flexible, and general method for semantic segmentation, termed Cross-Dataset Collaborative Learning (CDCL). Given multiple labeled datasets, we aim to improve the generalization and discrimination of feature representations on each dataset. Specifically, we first introduce a family of Dataset-Aware Blocks (DAB) as the fundamental computing units of the network, which help capture homogeneous representations and heterogeneous statistics across different datasets. Second, we propose a Dataset Alternation Training (DAT) mechanism to efficiently facilitate the optimization procedure. We conduct extensive evaluations on four diverse datasets, i.e., Cityscapes, BDD100K, CamVid, and COCO Stuff, with single-dataset and cross-dataset settings. Experimental results demonstrate our method consistently achieves notable improvements over prior single-dataset and cross-dataset training methods without introducing extra FLOPs. Particularly, with the same architecture of PSPNet (ResNet-18), our method outperforms the single-dataset baseline by 5.65%, 6.57%, and 5.79% of mIoU on the validation sets of Cityscapes, BDD100K, CamVid, respectively. Code and models will be released.

1. Introduction

Semantic segmentation has achieved significant improvement in recent years. The success is mainly attributed to better representations, well-designed frameworks, and the availability of diverse segmentation datasets. The most rapid progress comes from better representations learned by various deep networks, e.g., AlexNet [20], ResNet [15] and other variants. Semantic segmentation frameworks also have made great progress. Especially, FCN [24] achieves 65.3% mIoU on the Cityscapes [12] test set and it rises to 82.66% with DeepLabv3+ [11]. Multiple semantic segmentation datasets blossom (e.g., Cityscapes [12], COCO Stuff [6]), which provide rich data for supervised learning.

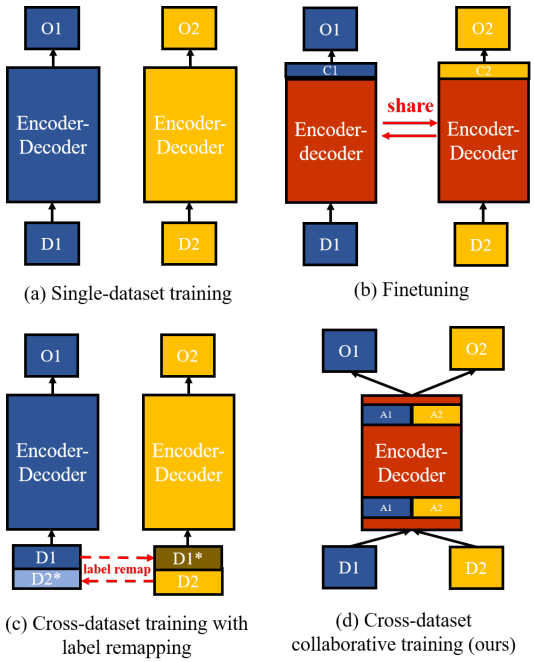


Figure 1: Illustration of prior methods and our cross-dataset collaborative learning for semantic segmentation. D is the dataset, O is the output, C is the classifier layer, and A is the dataset-specific layer.

From a perspective of data utilization, current methods are usually dataset-specific, i.e., trained and tested on a single dataset, as shown in Figure 1 (a). For further performance improvements, one straightforward way is to annotate more data for training. However, pixel-level manual annotations are time-consuming and labor-intensive[1]. Figure 1 (b) shows an effective approach. It first pretrains the

[1]For example, it takes about 90 minutes to label an image of the Cityscapes dataset [36].

network on the extra auxiliary datasets and then finetunes the feature representations on the target dataset. This finetuning approach provides a good initialization by leveraging extra data and sometimes can boost the performance of the target dataset. However, it is time-consuming which always requires to repeat the training process for different datasets and not applicable for joint optimization. It also needs to manually adjust some hyper-parameters (e.g., learning rate) for training. Figure 1 (c) presents another solution by training a unified model on a composite of multiple datasets. However, this approach requires to remap class labels from the other datasets to the target one, which is based on the assumptions of class overlap across different datasets. It often encounters inconsistent taxonomies and annotation practices, which may yield poor performance.

In this work, we focus on the fundamental problem of how to learn unified representations from various labeled datasets simultaneously for semantic segmentation, which has not been well investigated by the community. It remains challenging to build such a general system due to the inherent distribution shift across different datasets. To alleviate these problems, we propose a Cross-Dataset Collaborative Learning (CDCL) algorithm that is capable of learning from multiple datasets. First, we present an investigation of prior single-dataset and cross-dataset training baseline methods and analyze their limitations. Our key observations lie in: (1) The convolution filters can be shared for multiple datasets to maintain network efficiency. (2) The batch normalization layers are not appropriate to share across different datasets due to the bias of statistics. Second, motivated by the observations and analysis, we present a unified network to preserve the commonality and particularity of different datasets. Specifically, we introduce a Dataset-Aware Block (DAB) as the fundamental computing unit of the network, which helps capture homogeneous representations and heterogeneous statistics across different datasets. The proposed block is composed of a dataset-invariant convolution layer, multiple dataset-specific batch normalization layers, and an activation layer. As the convolution layers are shared across different datasets, our network does not introduce extra computation cost compared to the single-dataset baseline. Moreover, we propose a Dataset Alternation Training (DAT) mechanism to efficiently facilitate the optimization procedure. We conduct extensive experiments to demonstrate the effectiveness of the proposed method on diverse semantic segmentation benchmarks.

In summary, our main contributions include:

(1) We analyze the limitations of existing segmentation methods and explore the problem of the inability to directly use diverse datasets for network training.

(2) We propose a simple, flexible, and general cross-dataset collaborative learning algorithm, which can alleviate the distribution shift problem efficiently.

(3) We demonstrate the effectiveness of the proposed approach on diverse challenging datasets. Our method consistently achieves notable improvements over prior single-dataset and cross-dataset training baselines with the same computation budget.

2. Related Work

2.1. Semantic Segmentation

Semantic segmentation is a dense image prediction task, which plays a key role in high level scene understanding. FCN [24] is the pioneer of deep learning based semantic segmentation network, and subsequent methods have extended this base architecture. For example, PSPNet [35] and DeepLab [8, 9, 11] series adopt sophisticated feature extraction networks (e.g., ResNets [15] and DenseNets [17]) to learn discriminative feature representations for dense prediction. Besides, some critical strategies have been developed to further improve the performance, including atrous convolution [10], pyramid pooling module [35], attention mechanism [16, 13], context encoding [34], etc. In parallel, light-weight networks [25, 26, 27] become popular owing to their high speed and wide applications on resource-constrained devices. However, these networks are only trained on a single target dataset for fairly good performance, and can not maintain high accuracy on other datasets without finetuning.

2.2. Cross-Dataset Training

Recurrent Assistance [28] first proposed the cross-dataset training mechanism, where cross-dataset training is used for frame-based action recognition during the pretraining stage. Recent work explores cross-dataset training for object detection [32, 31]. [28] and [32] adopt a similar solution, where they proposed to generate a hybrid dataset by simple label concatenation or label mapping since the number and identity of classes are different in each dataset. Inspired by Squeeze-and-Excitation [16], the work of [31] introduced a domain attention module to make a single network active for universal object detection tasks.

Domain adaptation (DA) or knowledge transfer learning methods provide efficient technology for cross-dataset training, which aims to improve the performance of a target model with insufficient or lack of annotated data by using the knowledge from a source domain with adequate labeled data [37]. Learning domain-invariant representation is critical for the success of domain adaptation, especially for unsupervised DA. Many existing approaches, like [30, 4, 3] attempt to align data distributions either globally or locally to better achieve knowledge transfer from source to target domain. Different from DA, we treat all datasets as the target, and mainly study how to collaborative training from these datasets and improve the accuracy for each one.

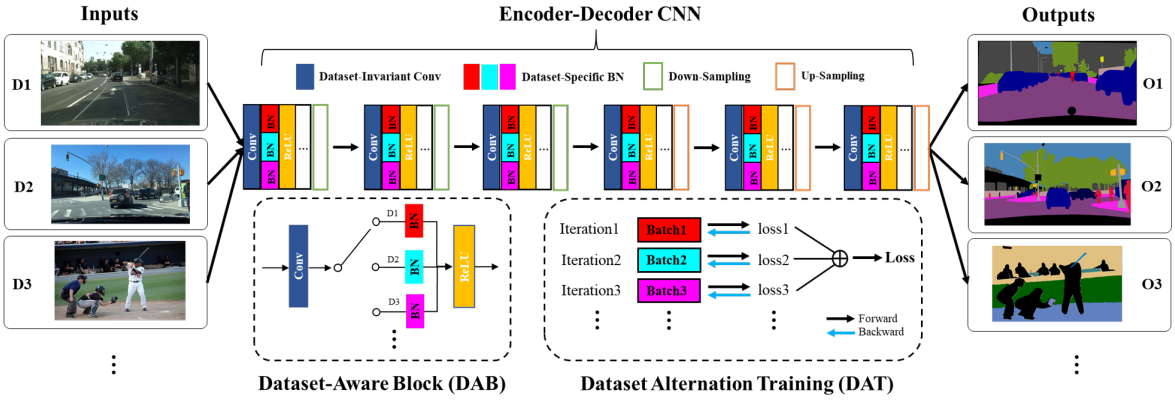


Figure 2: Overall structure of the proposed cross-dataset collaborative learning method. The input is a set of datasets. Dataset-Aware Block (DAB) is the fundamental computing unit of the encoder-decoder architecture, which consists of a dataset-invariant convolution layer, multiple dataset-specific batch normalization layers, and an activation layer. Dataset Alternation Training (DAT) is the proposed optimization strategy.

2.3. Multi-Task Learning

Another related research topic is multi-task learning (MTL) [1, 21, 14]. [1] jointly trained classification, detection, and semantic segmentation tasks on a single dataset. This method requires annotations for multiple tasks on a single dataset. [21] trained an integrated face analysis model by using multiple datasets annotated for different tasks (facial landmark, facial emotion, and face parsing), without the need of building a fully labeled common dataset for all the tasks. In multi-task learning methods, the performance is boosted by explicitly modeling the interaction of different tasks.

Our scenario is different from MTL. We aim to jointly train multiple datasets for a similar task. In this paper, we introduce a general framework for cross-dataset semantic segmentation. It can significantly improve the performance on various datasets simultaneously with the same FLOPs. Meanwhile, it also provides a feasible solution for practical applications.

3. Approach

In this section, we first analyze the most popular solutions for semantic segmentation. We then present our proposed cross-dataset collaborative learning framework and introduce each component in detail.

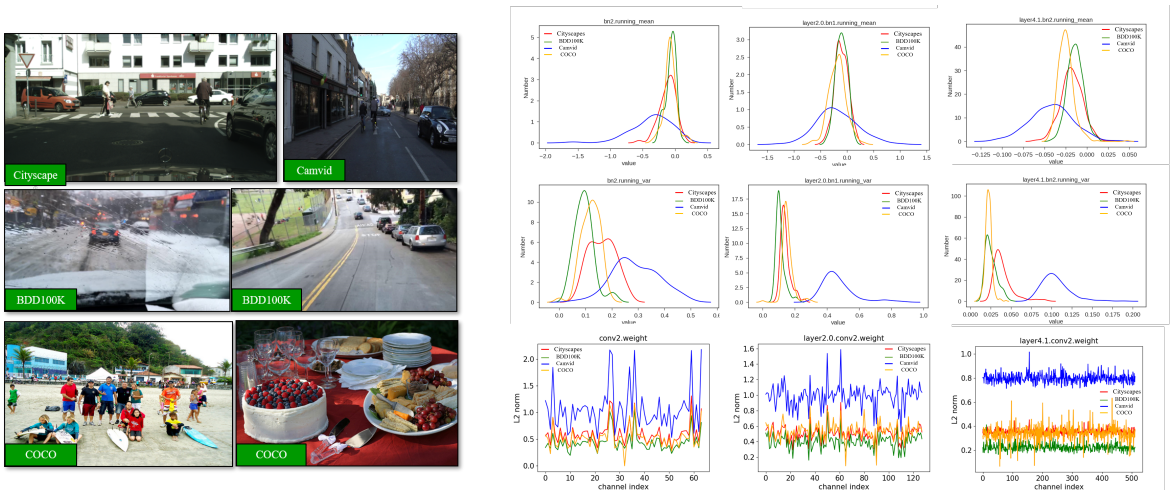
3.1. Analysis of Semantic Segmentation Methods

3.1.1 Single-Dataset Baseline

The encoder-decoder is used as the most popular architecture for semantic segmentation. It employs deep convolutional neural networks as the encoder to extract hierarchical features, and exploits a simple sub-network as the decoder to refine the segmentation results. Recent work (e.g., PSPNet [35] and DeepLab [11]) has extended this base architecture to improve segmentation performance on the target dataset. Towards different datasets, the simplest solution is to train and evaluate a network for each dataset separately. This solution is expensive since it needs to save all parameters of all networks and repeat the training process for each dataset independently.

3.1.2 Cross-Dataset Baseline

In addition to training the network on the specified dataset only, the finetuning based method is another straightforward way to improve the accuracy on the target dataset. Specifically, it first pretrains the network on the extra auxiliary datasets and then finetunes the feature representations on the target dataset. Using the auxiliary data sometimes is helpful to boost the performance for semantic segmentation. However, it is time-consuming which always requires to repeat the training process for different datasets and not applicable for joint optimization. It also needs to manually adjust some hyper-parameters (e.g., learning rate) for training. Another solution is to integrate multiple datasets with a similar scene and then train a network with the hybrid data. However, it requires prior knowledge of the target dataset because label mapping is always adopted to address the class duplication and conflict among different datasets. Moreover, this kind of dataset merging strategy is likely to yield poor performance due to the discrepancy of camera viewpoints, scenes, etc.



(a) Visualization for samples of different datasets

(b) The statistic distributions of BN and Conv for different datasets.

Figure 3: Comparisons of different datasets. (a) Visualization for samples of different datasets. (b) The distributions of BN and Conv of single-dataset semantic segmentation networks on four datasets. First row: running mean of BN. Second row: running variance of BN. Third row: L2 norm of Conv weights. Here, we take two low-level layers (*bn2* and *layer2*) and one high-level layer (*layer4*) of PSPNet (ResNet-18) network as examples to show the results.

3.2. Cross-Dataset Collaborative Learning

3.2.1 Rethinking BatchNorm and Convolution

To explore how to effectively alleviate the above dilemmas, we analyze the distribution shift problem among different semantic segmentation datasets. First, Figure 3 (a) shows example images from multiple datasets, we can see that different segmentation datasets vary greatly (e.g., scenes and illumination). It is challenging to directly training on multiple datasets. Second, we analyze the parameter distributions of convolution and batch normalization layers of four networks which are trained on four different datasets (i.e., Cityscapes, BDD100K, CamVid, and COCO) separately. Figure 3 (b) presents these parameter distributions. For Conv, weights hold the same distributions, which implies that the Conv layers can be shared across datasets. For BN, we observe that the distributions of both running mean and running var have different shapes for different datasets. For example, Cityscapes and BDD100K have similar scenes and share the same label space, but their BN distributions are significantly different. This is because BN [18] whitens activations within a mini-batch and transforms the whitened activations using affine parameters. Both of the whitening and transformation operations are sensitive to the dataset itself. Specifically, during training, BN estimates the mean and variance of the input activations through the exponential moving average method. In the testing phase, BN uses the estimated mean and variance for whitening activations and uses the learned affine transformation parameters to recover the representation ability of the activations. Thus, directly sharing these BN parameters for different datasets is inappropriate due to the inherent distribution shift.

3.2.2 Dataset-Aware Block

Inspired by the above analysis, we propose a simple and flexible framework for cross-dataset collaborative learning for segmentation. As shown in Figure 2, we apply a dataset-aware block (DAB) as the fundamental computing unit of the encoder-decoder architecture. The DAB consists of a dataset-invariant convolution layer, multiple dataset-specific batch normalization layers, and an activation layer. Figure 2 illustrates how the DAB works. Specifically, we apply a dataset-shared convolution layer to extract the representations. Then we use the dataset-specific batch normalization layer to whiten the activations following the same principle as [7]. In each block, the number of our BN layers is identical to the number of datasets and each BN layer is responsible for its corresponding dataest. A switch is automatically to determine which BN should be activated based on the data source. Suppose we have N datasets in total: $D_{i}{}_{i=1}^{N}$, our dataset-specific BN can be formulated by:

$$\text{DSBN}\{D_i\}(X_i;\gamma_i,\beta_i) = \gamma_i\hat{X}_i + \beta_i \tag{1}$$

where,

$$\begin{aligned} \hat{X}_i &= \frac{X_i-\mu_i}{\sqrt{\sigma_i^2+\epsilon}} \\ \mu_i = \frac{1}{B}\sum_{j=1}^{B} X_i^j,\ \sigma_i^2 &= \frac{1}{B}\sum_{j=1}^{B}(X_i^j-\mu_i)^2 \end{aligned} \tag{2}$$

Here, μ_i and σ_i^2 denote the empirical mean and variance, respectively, γ_i and β_i denote the affine transformation pa-

rameters, B denotes the batch size, and ϵ is a small constant added for numerical stability.

We expect each BN to capture the specific dataset information by estimating batch statistics and learning affine parameters separately. By constructing dataset-shared Conv and dataset-specific BN layers, our DAB can help capture homogeneous representations and heterogeneous statistics across different datasets. Besides, we also exploit dataset-aware classifiers for final predictions since different datasets have different categories.

3.2.3 Dataset Alternation Training

To reduce the training instability caused by the discrepancy of different feature map distributions, we introduce a dataset alternation training (DAT) mechanism for efficiently optimizing the network in the cross-dataset setting. As shown in Figure 2, in each iteration, we construct the batch with samples from a single dataset and compute the corresponding loss by forward propagation. In the next iteration, we select samples from another dataset to build the batch in an alternating manner. After obtaining the losses for each dataset, we accumulate them and backpropagate the entire gradients for each dataset flow. We repeat such training procedures for network convergence. We also try larger iteration interval t for DAT, which means training a dataset t times and then training another one time. We find $t = 1$ works best and detailed ablation results are presented in the following experiments. We summarize the merits of DAT in two aspects. First, backpropagating the loss of each dataset in each iteration will incur training instability. DAT can mitigate this problem by optimizing the entire loss computed from different datasets. Second, DAT provides an efficient way to train the samples from multiple datasets instead of putting them into a large batch.

We follow the baseline method to use the conventional multi-class cross-entropy loss for training. Our final objective function for cross-dataset training can be formulated as below and can be minimized end-to-end.

$$L = -\sum_{i=1}^{N}\sum_{j=1}^{M} w^i y_j^i log(p_j^i) \tag{3}$$

where, N denotes the number of datasets, M means the number of image pixels, p_j^i and y_j^i refer the predicted probability and corresponding label for the j-th pixel on the i-th dataset, respectively. w^i denotes the loss weight and we set $w^i = 1$ to make these loss value ranges comparable.

4. Experiments

In this section, we conduct extensive experiments and detailed ablation studies to validate the effectiveness of our cross-dataset collaborative learning method on diverse datasets.

4.1. Experimental Setting

4.1.1 Datasets

Cityscapes. The Cityscapes dataset [12] is collected for urban scene understanding. The dataset contains 5,000 high-quality finely annotated images and 20,000 coarsely annotated images. The finely annotated images are divided into 2,975, 500, 1,525 images for training, validation, and testing. We only use the finely annotated dataset in our experiments and perform detailed comparison experiments on its validation set and test set.

BDD100K. BDD100K [33] is a large-scale diverse driving video database with 7,000 training and 1,000 validation images captured at different weather conditions as well as different times of the day. We evaluate the performance on the validation set using 19 classes that are the same with Cityscapes.

CamVid. CamVid [5] is another automotive dataset. For a fair comparison with prior work, we adopt the split of Paszker et al. [27], which partitions the dataset into 367, 100, and 233 images for training, validation and testing respectively. We use 11 classes for evaluation.

COCO Stuff. COCO Stuff [6] is a challenging dataset for segmentation in general scenarios. It includes 164K images from COCO 2017 (train 118K, val 5K, test-dev 20K) and contains up to 182 categories (91 thing classes and 91 stuff classes). We use all the categories for evaluation.

4.1.2 Implementation Details

In our experiments, we use the architecture of PSPNet with the pretrained ResNet-18 or ResNet-101 on ImageNet as the segmentation baseline. We use ResNet-18 as backbone for all ablation studies. The networks are trained using mini-batch stochastic gradient descent (SGD) with momentum of 0.9, weight decay of 0.0001, and a batch size of 8. The initial learning rate is set to 0.01 and multiplied by $(1 - \frac{iter}{maxiter})^{0.9}$ with a polynomial decaying policy. We randomly crop the images into 512×512 and use standard data augmentations for training, such as random scaling (0.5 $\sim$ 2.1) and random flipping. We use the metric of mean IoU (mIoU) to evaluate the segmentation accuracy for each dataset.

4.2. Results

We compare the proposed cross-dataset collaborative learning algorithm with three baseline methods in Table 1 (a-c). We conduct experiments using four different cross-dataset settings, including Cityscapes + BDD100K, Cityscapes + CamVid, Cityscapes + COCO, and Cityscapes + BDD100K + CamVid. For fair comparisons, we use the same network structure and basic loss function as baselines.

Cityscapes + BDD100K. In the cross-dataset setting of Cityscapes + BDD100K, the two datasets share the same categories and similar scenes. Table 1 (a) shows that compared with the singe-dataset and finetuning baselines, our method can improve mIoU on Cityscapes by 5.11% and 4.84%, respectively. In addition, our method simplifies the training scheme which does not need to repeat the training process multiple times. Notably, the mIoU based on label remapping for Cityscapes suffers a minor drop from 67.52% to 66.23%, which can be explained by the fact that simply concatenating two datasets may cause disturbation for Cityscapes segmentation. In contrast, our method effectively eases dataset bias and bring accuracy gains of 6%. We draw a similar conclusion on BDD100K. In the case of two datasets having the same classes, the results demonstrate that our method can sufficiently utilize the complementarity between different datasets to improve accuracy on each dataset.

Cityscapes + CamVid. In the cross-dataset setting of Cityscapes + CamVid, the two datasets share similar scenes but have different label spaces. The larger Cityscapes dataset has 19 classes and the smaller CamVid dataset has 11 classes. Table 1 (b) presents the results of the different segmentation solutions. Taking CamVid as the target dataset, the mIoU performance can be improved from 73.05% to 74.83% by pretraining on Cityscapes and finetuning on CamVid. Label remapping method achieves 4.5% mIoU improvement compared with the single-dataset baseline (73.05%), which is an efficient method to alleviate the low performance by training the network on a hybrid dataset. Compared with the above methods, our method performs the best both on CamVid and Cityscapes without the complex training or label prepossessing scheme. The results demonstrate that our cross-dataset collaborative learning method is also effective for datasets with different categories.

Cityscapes + COCO. In the cross-dataset setting of Cityscapes + COCO, the two datasets have different scenes and different label spaces. The Cityscapes dataset contains 19 classes of automotive scenes, and the COCO dataset has 182 categories of general scenarios. Besides, the COCO dataset has 118k images and it is 39 times that of Cityscapes. Currently, some segmentation models always use COCO as a pretraining dataset to boost the performance on Cityscapes. By cross-dataset training on Cityscapes and COCO, our goal is a single model that has the same backbone and can segment 182 classes on COCO without accuracy loss, and improve the Cityscapes accuracy as well. Table 1 (c) shows that the proposed collaborative learning method gains 5.1% and 2.2% improvements on Cityscapes compared with single-dataset and finetuning baselines, respectively. The results prove that our method can effectively process two datasets with different scenes and a huge discrepancy of data volume.

Method	Cityscapes (%)		BDD100K(%)
	Val.	Test	Val.
Single-dataset	67.52	67.75	53.88
Finetuning	67.79	66.52	58.30
Label remapping	66.23	66.39	58.74
CDCL (Ours)	**72.63**	**71.55**	**60.47**

(a) Cityscapes + BDD100K

Method	Cityscapes (%)		CamVid (%)	
	Val.	Test	Val.	Test
Single-dataset	67.52	67.75	73.05	70.41
Finetuning	67.35	67.87	74.83	71.16
Label remapping	67.13	68.22	78.03	76.86
CDCL (Ours)	**69.77**	**68.56**	**78.45**	**77.34**

(b) Cityscapes + CamVid

Method	Cityscapes (%)		COCO (%)
	Val.	Test	Val.
Single-dataset	67.52	67.75	32.86
Finetuning	70.44	70.23	32.10
Label remapping	50.35	52.39	32.67
CDCL (Ours)	**72.63**	**72.52**	**32.87**

(c) Cityscapes + COCO

Table 1: Performance comparisons between existing baselines and our method with the same ResNet-18 backbone in three different cross-dataset settings.

Cityscapes + BDD100K + CamVid. We also conduct experiments in the three-dataset setting where Cityscapes, BDD100K and CamVid are used. Observation from Table 2 suggests that our method enables a gain of $\geq$5 points on all three datasets. Furthermore, we also analyze the IoU for each category and provide detailed information in the supplementary material. Figure 4 presents the per-class IoU comparison between the single-dataset method and our solution on Cityscapes and CamVid. It indicates that our cross-dataset training outperforms the prior method both for some overlapped categories (e.g., rider and traffic sign) and some unique classes, like the motorbike in Cityscapes. The experiments above demonstrate the effectiveness of cross-dataset collaborative learning in general semantic segmentation. By cross-dataset training, we can utilize the existing datasets to improve the accuracy without extra works, such as network retraining, datasets alignment, or even time-consuming and laborious collection and labeling of a large-scale and diverse dataset.

4.3. Ablation Study

Effects of DAB and DAT. The fact that Cityscapes and BDD100K have the same label space and very similar assignment distributions, suggests a substantial domain over-

Method	Cityscapes (%)		CamVid (%)		BDD100K (%)
	Validation	Test	Validation	Test	Validation
Single-dataset	67.52	67.75	73.05	70.41	53.88
CDCL (Ours)	73.17 **(+5.65)**	70.98 **(+3.23)**	78.84 **(+5.79)**	75.52 **(+5.11)**	60.45 **(+6.57)**

Table 2: Performance comparisons between the single-dataset baseline and our method with the same ResNet-18 backbone in the three-dataset setting.

Method	Norm	DAT	Cityscapes	BDD100K
Single-dataset	BN		67.75%	53.88%
Cross-dataset	BN		62.10%	53.34%
Cross-dataset	BN	✓	64.69%	56.33%
Cross-dataset	DSBN		68.55%	58.93%
CDCL (Ours)	DSBN	✓	**72.63%**	**60.47%**

Table 3: Ablation studies on DSBN and DAT with the ResNet-18 backbone on the Cityscapes and BDD100K validation sets.

Iteration interval	Cityscapes (%)	BDD100K (%)
$t = 1$	72.63	60.47
$t = 2$	68.88	59.17
$t = 3$	65.11	57.79
$t = 4$	62.86	57.21
$t = 5$	61.12	56.06

Table 4: Effect of iteration interval t in DAT on the Cityscapes and BDD100K validation sets.

Method	Cityscapes (%)		BDD100K (%)
	Val.	Test	Val.
Single-dataset	73.51	74.45	57.47
CDCL (Ours)	**75.83**	**75.95**	**63.84**

(a) Cityscapes + BDD100K

Method	Cityscapes (%)		CamVid (%)	
	Val.	Test	Val.	Test
Single-dataset	73.51	74.45	75.86	74.72
CDCL (Ours)	**75.00**	**74.77**	**81.15**	**79.33**

(b) Cityscapes + CamVid

Table 5: Performance comparisons between the single-dataset baseline and our method with the same ResNet-101 backbone in the two-dataset setting.

lap. Thus we conduct experiments on these two datasets to investigate the effect of dataset-specific BN. For a fair comparison, we provide the performance of keeping BN shared across two datasets. Table 3 shows that DSBN significantly outperforms the shared BN method by $1 \sim 3.6\%$. This is not surprising, given the above discussed inadequacy of shared BN to solving the distribution shift between two datasets, which even share the same label space.

Meanwhile, to assess the efficiency of the dataset alternation training strategy, we compare the performance under the shared BN and DSBN settings. Taking cityscapes as an example, only adding the DAT can boost the mIoU by 2.5%, however, it is still much weaker than that of with single-dataset method. Finally, the cross-dataset method has the best performance by combining Dataset-aware Block with DAT (4.55% and 6.59% points on Cityscapes and BDD100K, respectively), which further verifies the effectiveness of the proposed method.

Parameter Analysis for DAT. Table 4 summarizes how the performance of the dataset alternation training depends on iteration interval t. As iteration interval t increases, the accuracy suffers due to the weights of convolution layers change frequently. For simplicity, we again use Cityscapes and BDD100K in this experiment. When $t = 1$, we sum the losses of Cityscapes and BDD100K at each iteration and then perform backward once, when $t = 2$, more examples ($2\times$) from BDD100K are utilized for parameters updating, leading to weights are more suitable for BDD100K and perform worst on Cityscapes. Thus, an iteration interval size of $t = 1$ is suggested which exhibits stable performance both on Cityscapes and BDD100K.

Effect of Backbone. We conduct experiments on another backbone, i.e. ResNet-101, to illustrate that our cross-dataset learning method works based on the backbone with redundant parameters. Results in Table 5 (a-b) support our assumption. Table 5 shows that the proposed cross-dataset training setting achieves promising performance on all datasets, indicating that our training pipeline is a universal solution, not depending on the backbone network.

4.4. Comparisons to the State-of-the-Arts

Table 6 presents the segmentation accuracy on Cityscapes test set with state-of-the-art semantic segmentation networks, we use FLOPs to evaluate the network

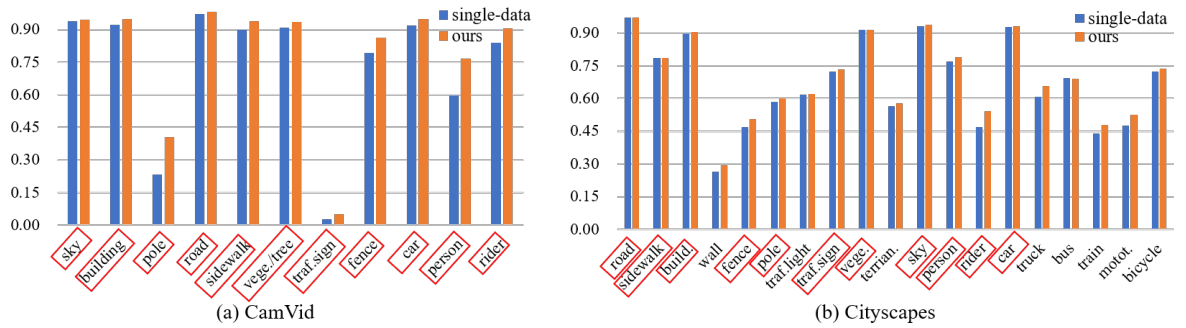


Figure 4: Quantitative analysis of per-class accuracy on (a) CamVid and (b) Cityscapes validation sets. The red box means the overlapped categories between Cityscapes and CamVid.

Method	GFLOPs	Cityscapes (%)
Current state-of-the-art results		
SegNet [2]	286.0	56.10
ENet [27]	7.6	58.30
ESPNetv2 [26]	5.4	65.10
ESPNet [25]	8.9	60.30
ERFNet [29]	25.6	68.00
FCN-8s [24]	1335.6	65.30
RefineNet [22]	2102.8	73.60
Results w/o and w/ our scheme		
PSPNet (ResNet-18) [23]	512.8	67.60
PSPNet (ResNet-18)† [23]	512.8	71.40
PSPNet (ResNet-18) (Ours)	512.8	71.00
PSPNet (ResNet-18) (Ours)‡	1730.7	72.52
PSPNet (ResNet-101) [23]	2299.8	77.60
PSPNet (ResNet-101) (Ours)*	2299.8	78.74
PSPNet (ResNet-101)*‡	7762.0	78.40
PSPNet (ResNet-101) (Ours)*‡	7762.0	**79.73**

Table 6: Comparisons with the state-of-the-art methods on the Cityscapes test set. $*$: refers training with random crop of 769×769. †: refers using knowledge distillation method in [23]. ‡: refers testing with multiple scales.

complexity which is calculated at the same image resolution used for computing the accuracy. With collaborative training with BDD100K, our approach can improve the results over different complexity networks: PSPNet (ResNet-18) and PSPNet (ResNet-101). For the network with a light-weight backbone such as PSPNet (ResNet-18), the improvement is significant with 3.4% on the test set compared with the baseline of 67.6%. We also do a comparison with state-of-the-art methods on CamVid test set, Table 7 shows our method can significantly improve mIoU by 5.0% compared with [23], which uses an extra 2000 unlabeled images collected from the Cityscapes dataset.

Method	Extra data	CamVid
Current state-of-the-art results		
ENet [27]	no	51.3%
FC-DenseNet56 [19]	no	58.9%
SegNet [2]	ImN	55.6%
DeepLab-LFOV [9]	ImN	61.6%
FCN-8s [24]	ImN	57.0%
ESPNet-C† [23]	unl	64.1%
ESPNet† [23]	unl	65.1%
Results w/o and w/ our scheme		
PSPNet (ResNet-18) [23]	ImN	70.3%
PSPNet (ResNet-18)† [23]	ImN	71.0%
PSPNet (ResNet-18)† [23]	ImN+unl	72.3%
PSPNet (ResNet-18) (Ours)	ImN+Cityscapes	**77.3%**

Table 7: Comparisons with the state-of-the-art methods on the CamVid test set. ImN: ImageNet dataset. unl: unlabeled street scene dataset sampled from Cityscapes. †: using knowledge distillation method in [23].

Compared to the distillation method, we provide another effective pipeline which can utilize the existing datasets to improve the classes without extra work of labeling, to achieve higher performance with the same FLOPs. We also provide some qualitative results of our proposed method on different datasets in Figure 5 to show the effectiveness.

5. Conclusion

In this work, we build a unified cross-dataset collaborative learning segmentation algorithm that is capable of learning from multiple datasets. It can help capture homogeneous representations and heterogeneous statistics across different datasets. The proposed method outperforms current segmentation methods on diverse challenging datasets with the same computation cost and provides a feasible so-

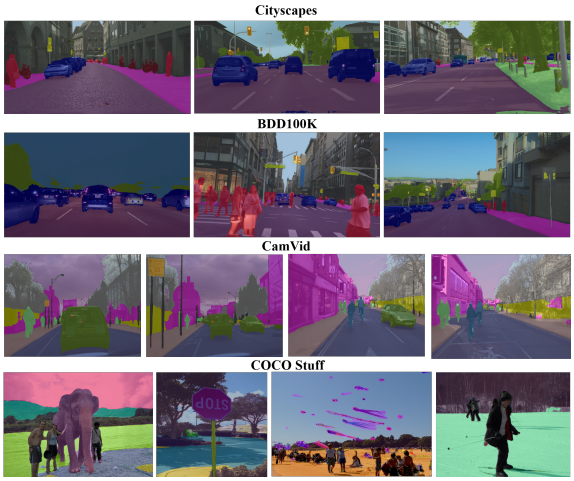


Figure 5: Qualitative results on four datasets, produced from PSPNet (ResNet-18) with our provided method.

lution for practical applications.

References

[1] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-task feature learning. *NeurIPS*, 19:41–48, 2006. 3

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *TPAMI*, 39(12):2481–2495, 2017. 8

[3] Matteo Biasetton, Umberto Michieli, Gianluca Agresti, and Pietro Zanuttigh. Unsupervised domain adaptation for semantic segmentation of urban scenes. In *CVPRW*, pages 0–0, 2019. 2

[4] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *CVPR*, pages 3722–3731, 2017. 2

[5] Gabriel J Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2):88–97, 2009. 5

[6] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *CVPR*, pages 1209–1218, 2018. 1, 5

[7] Woong-Gi Chang, Tackgeun You, Seonguk Seo, Suha Kwak, and Bohyung Han. Domain-specific batch normalization for unsupervised domain adaptation. In *CVPR*, pages 7354–7362, 2019. 4

[8] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014. 2

[9] 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. *TPAMI*, 40(4):834–848, 2017. 2, 8

[10] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2

[11] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, pages 801–818, 2018. 1, 2, 3

[12] 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*, pages 3213–3223, 2016. 1, 5

[13] Jun Fu, Jing Liu, Yuhang Wang, Yong Li, Yongjun Bao, Jinhui Tang, and Hanqing Lu. Adaptive context network for scene parsing. In *CVPR*, pages 6748–6757, 2019. 2

[14] Georgia Gkioxari, Bharath Hariharan, Ross Girshick, and Jitendra Malik. R-cnns for pose estimation and action detection. *arXiv preprint arXiv:1406.5212*, 2014. 3

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 2

[16] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, pages 7132–7141, 2018. 2

[17] Forrest Iandola, Matt Moskewicz, Sergey Karayev, Ross Girshick, Trevor Darrell, and Kurt Keutzer. Densenet: Implementing efficient convnet descriptor pyramids. *arXiv preprint arXiv:1404.1869*, 2014. 2

[18] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 4

[19] Simon Jégou, Michal Drozdzal, David Vazquez, Adriana Romero, and Yoshua Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *CVPRW*, pages 11–19, 2017. 8

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 1

[21] Jianshu Li, Shengtao Xiao, Fang Zhao, Jian Zhao, Jianan Li, Jiashi Feng, Shuicheng Yan, and Terence Sim. Integrated face analytics networks through cross-dataset hybrid training. In *ACM MM*, pages 1531–1539, 2017. 3

[22] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR*, pages 1925–1934, 2017. 8

[23] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. Structured knowledge distillation for semantic segmentation. In *CVPR*, pages 2604–2613, 2019. 8

[24] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015. 1, 2, 8

[25] Sachin Mehta, Mohammad Rastegari, Anat Caspi, Linda Shapiro, and Hannaneh Hajishirzi. Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation. In *ECCV*, pages 552–568, 2018. 2, 8

[26] Sachin Mehta, Mohammad Rastegari, Linda Shapiro, and Hannaneh Hajishirzi. Espnetv2: A light-weight, power efficient, and general purpose convolutional neural network. In *CVPR*, pages 9190–9200, 2019. 2, 8

[27] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*, 2016. 2, 5, 8

[28] Toby Perrett and Dima Damen. Recurrent assistance: cross-dataset training of lstms on kitchen tasks. In *CVPRW*, pages 1354–1362, 2017. 2

[29] Eduardo Romera, José M Alvarez, Luis M Bergasa, and Roberto Arroyo. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 19(1):263–272, 2017. 8

[30] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*, pages 3723–3732, 2018. 2

[31] Xudong Wang, Zhaowei Cai, Dashan Gao, and Nuno Vasconcelos. Towards universal object detection by domain attention. In *CVPR*, pages 7289–7298, 2019. 2

[32] Yongqiang Yao, Yan Wang, Yu Guo, Jiaojiao Lin, Hongwei Qin, and Junjie Yan. Cross-dataset training for class increasing object detection. *arXiv preprint arXiv:2001.04621*, 2020. 2

[33] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687*, 2(5):6, 2018. 5

[34] Hang Zhang, Kristin Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Ambrish Tyagi, and Amit Agrawal. Context encoding for semantic segmentation. In *CVPR*, pages 7151–7160, 2018. 2

[35] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, pages 2881–2890, 2017. 2, 3

[36] 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 *NeurIPS*, pages 7287–7300, 2019. 1

[37] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, pages 289–305, 2018. 2