Drive&Segment: Unsupervised Semantic Segmentation of Urban Scenes via Cross-modal Distillation – Supplementary material –

Antonin Vobecky[1,2], David Hurych[2], Oriane Siméoni[2], Spyros Gidaris[2], Andrei Bursuc[2], Patrick Pérez[2], and Josef Sivic[1]

[1] Czech Institute of Informatics, Robotics and Cybernetics, CTU in Prague
[2] valeo.ai

A Additional quantitative results

A.1 Results with another training dataset

We report in Tables 4 and 5 the performance of Drive&Segment when trained using a subset of ∼8k images from the nuScenes dataset [1]. As shown in Table 4, the mIoU on Cityscapes is 19.8. Although there is a small drop from the 21.8 achieved with Drive&Segment trained on Waymo Open, the results are still significantly better than those of the competing methods. This drop might be caused by differences in statistics between the two datasets, *e.g.*, nuScenes has fewer examples of smaller-object classes, such as pedestrians.

A.2 Ablation of the number of clusters in unsupervised labeling

Here we investigate the sensitivity of our method to the number k of clusters used for unsupervised labeling. Figure 7 shows the mIoU results on Cityscapes for $k \in \{20, 25, 30, 35, 40\}$. In all cases, we use a ViT-S/16 feature extractor trained with DINO. The results show that for $k \in \{20, 25, 30, 35\}$ the *mIoU performance is fairly stable*. As expected, when the number of clusters becomes much higher than the number of Cityscapes classes (*e.g.*, $k = 40$), the performance drops.

A.3 Influence of the LiDAR's density

We investigate here the performance of Drive&Segment when provided with sparser LiDAR data. We performed experiments on the Waymo Open dataset and downsampled the LiDAR data from 64 to 32 beam channels by dropping every other channel. We re-trained the *teacher* model three times and report the average performance (following the setup of the main paper). We obtained 20.3 mIoU, which is only slightly lower than the 20.4 obtained with the full LiDAR resolution, demonstrating the robustness of our method to this considerable decrease of LiDAR resolution. However, as already discussed in the main

Table 4. Comparative results of unsupervised semantic segmentation methods when trained on nuScenes. Comparison to the state of the art on Cityscapes [4] (CS), DarkZurich [9] (DZ) and Nighttime Driving [5] (ND) datasets measured by the mean IoU (mIoU). The colored differences are reported with respect to the state-of-the-art approach of [3] denoted by ⚓; '*sup. init.*' stands for supervised initialization of the *encoder* and the column '*train. data*' indicates the dataset used for training, namely nuScenes [1] (nuSC).

architecture, method	sup. init.	train. data	CS19 [4] mIoU	CS27 [4] mIoU	DZ [9] mIoU	ND [5] mIoU
RN18+FPN						
⚓ PiCIE* [3]	yes	nuSC	15.8	9.7	4.6	9.9
Modified DC* [2]	yes	nuSC	11.6 (-4.2)	7.1 (-2.6)	7.7 (+3.1)	8.3 (-1.6)
Drive&Segment (Ours, S)	yes	nuSC	16.2 (+0.4)	11.4 (+1.7)	7.5 (+2.9)	10.2 (+0.3)
Segmenter, ViT-S/16						
Drive&Segment (Ours, S)	no	nuSC	19.8 (+4.0)	13.9 (+4.2)	9.7 (+5.1)	14.1 (+4.2)

* Our training using PiCIE code base.

Table 5. Comparative results on ACDC when methods trained on nuScenes. Comparison to the state of the art for unsupervised semantic segmentation on the ACDC [10] dataset. Please refer to Table 4 for the symbols.

architecture, method	sup. init.	train. data	night mIoU	fog mIoU	rain mIoU	snow mIoU	average mIoU
RN18+FPN							
⚓ PiCIE* [3]	yes	nuSC	4.3	8.9	9.5	7.5	7.5
Modified DC* [2]	yes	nuSC	6.7 (+2.4)	11.7 (+2.8)	10.4 (+0.9)	9.6 (+2.1)	9.6 (+2.1)
Drive&Segment (Ours, S)	yes	nuSC	7.9 (+3.6)	14.3 (+5.4)	14.4 (+4.9)	13.4 (+5.9)	12.5 (+5.0)
Segmenter, ViT-S/16							
Drive&Segment (Ours, S)	no	nuSC	10.6 (+6.3)	13.3 (+4.4)	16.0 (+6.5)	14.8 (+7.3)	13.9 (+6.4)

paper, our method will likely not work well with extremely sparse LiDAR data (*e.g.*, low-cost LiDARs with 4-beam channels). Such a sparsity would lead to poor LiDAR-based segments and geometric priors that would rather confuse the model, instead of teaching it to recognize objects.

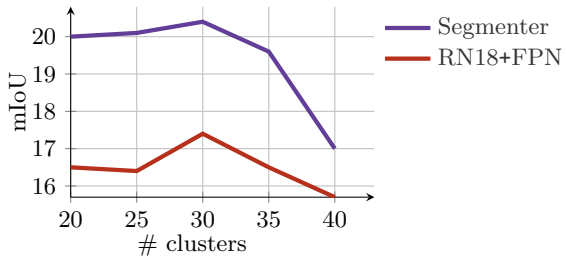


Fig. 7. Ablation of the number of clusters. Performance in mIoU, when using the **Segmenter** model and the **ResNet18+FPN** model on the Cityscapes dataset, as a function of the number of clusters in the unsupervised labeling step.

Table 6. Comparative results using PA metric. Comparison to the state of the art for unsupervised semantic segmentation on Cityscapes [4] (CS), DarkZurich [9] (DZ) and Nighttime driving [5] (ND) datasets measured by the pixel accuracy (PA). Same organization as Table 4. For easy reference, rows are colored according to the used training dataset.

architecture, method	sup. init.	train. data	CS19 [4] PA		CS27 [4] PA		DZ [9] PA		ND [5] PA	
RN18+FPN										
⚓ PiCIE‡ [3]	yes	CS	63.1		62.7		30.7		41.4	
IIC† [8]	yes	CS		-	47.9	(-14.8)		-		-
Modified DC‡ [2]	yes	CS	52.4	(-10.7)	52.1	(-10.7)	42.4	(+11.7)	46.2	(+4.8)
Modified DC⋆	yes	nuSc	45.9	(-17.2)	45.7	(-17.0)	41.4	(+10.7)	41.9	(+0.5)
PiCIE⋆	yes	nuSc	61.6	(-1.5)	61.3	(-1.4)	29.6	(-1.1)	45.1	(+3.7)
Drive&Segment (Ours, S)	yes	nuSc	61.4	(-1.7)	61.1	(-1.6)	37.4	(+6.7)	33.6	(-7.8)
Modified DC⋆	yes	WO	55.6	(-7.5)	43.2	(-19.5)	35.8	(+5.1)	33.4	(-8.0)
PiCIE⋆	yes	WO	48.6	(-14.5)	48.3	(-14.4)	31.9	(+1.1)	40.0	(-1.4)
Drive&Segment (Ours, S)	yes	WO	66.4	(+3.3)	67.1	(+4.3)	47.7	(+17.0)	49.0	(+7.6)
Segmenter, ViT-S/16										
Drive&Segment (Ours, S)	no	nuSc	73.2	(**+10.1**)	72.8	(**+10.1**)	50.2	(+19.5)	65.5	(**+24.1**)
Drive&Segment (Ours, S)	no	WO	**69.5**	(+6.4)	**69.1**	(+6.4)	**55.9**	(+25.1)	**60.2**	(+18.8)

† Results reported in [3]. ‡ Models provided by the PiCIE [3] authors.
⋆ Trained by PiCIE code base.

Table 7. Comparative results on ACDC using PA metric. Comparison to the state-of-the-art approach [3] for unsupervised semantic segmentation on the ACDC [10] dataset. Same organization as Table 5. For easy reference, rows are colored according to the used training dataset

method	sup. init.	train. data	night PA		fog PA		rain PA		snow PA		average PA	
RN18+FPN												
⚓ PiCIE [3]	yes	CS	25.8		50.0		53.6		50.4		45.0	
MDC [2]	yes	CS	43.0	(+17.3)	43.6	(-6.4)	35.0	(-18.6)	38.8	(-11.5)	40.1	(-4.8)
Modified DC⋆	yes	nuSC	36.5	(+10.7)	44.8	(-5.2)	41.4	(-12.2)	38.5	(-11.9)	40.3	(-4.7)
PiCIE⋆	yes	nuSC	26.9	(+1.1)	33.1	(-16.9)	33.4	(-20.2)	29.1	(-21.3)	30.6	(-14.4)
Drive&Segment (Ours, S)	yes	nuSC	34.5	(+8.7)	59.4	(+9.4)	58.2	(+4.6)	53.9	(+3.5)	51.5	(+6.5)
MDC⋆	yes	WO	32.9	(+7.2)	47.0	(-3.0)	40.3	(-13.3)	44.2	(-6.2)	41.1	(-3.8)
PiCIE⋆	yes	WO	27.2	(+1.4)	56.9	(+6.8)	53.8	(+0.2)	53.0	(+2.6)	47.7	(+2.8)
Drive&Segment (Ours, S)	yes	WO	43.2	(+17.5)	56.5	(+6.5)	54.1	(+0.5)	55.5	(+5.1)	52.3	(+7.4)
Segmenter, ViT-S/16												
Drive&Segment (Ours, S)	no	nuSC	50.2	(+24.4)	60.2	(+10.2)	62.5	(+8.9)	56.5	(+6.1)	57.5	(+12.5)
Drive&Segment (Ours, S)	no	WO	**52.6**	(+26.9)	54.2	(+4.2)	50.1	(-3.5)	**56.8**	(+6.4)	**53.4**	(+8.5)

A.4 Pixel accuracy results

In Tables 6 and 7, we report results measured with the pixel accuracy (PA) metric corresponding to all experiments of our main paper. We observe that results follow a similar trend to those measured with mIoU.

A.5 Category-wise results

In the main paper, we have presented results averaged over all classes. We report in Table 8 the *per-class IoU* results of our Drive&Segment approach on the Cityscapes dataset.

Table 8. Per-class comparative performance on Cityscapes. Per-class IoU is evaluated using the Hungarian algorithm on the 19 validation classes. We can see significant benefits of Drive&Segment ('D&S') over PiCIE in 14 (including all road users and objects) out of 19 classes. Drive&Segment works much worse for *sidewalk* and *sky* as we discuss in Sections A.5 and B.4. '`(CS)`' stands for a model trained on the Cityscapes [4] dataset, while '`(WO)`' for models trained on the Waymo Open [11] dataset. The best results per class are highlighted in bold and color.

	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorcycle	bicycle	mIoU
RN18+FPN																				
PiCIE [3] (`CS`)	58.2	12.5	63.8	1.0	2.4	1.3	0.1	0.4	55.5	1.7	44.7	1.9	0.5	48.2	1.3	3.9	1.0	0.5	1.6	15.8
PiCIE [3] (`WO`)	58.5	**13.8**	35.8	**6.7**	0.7	1.2	0.4	1.2	28.3	1.2	**55.8**	3.1	0.6	48.5	0.5	1.5	0.3	0.0	2.3	13.7
D&S (Ours, `WO`)	72.7	7.0	56.6	4.5	**5.6**	16.9	3.6	15.7	**66.8**	**2.2**	6.0	40.0	**5.0**	44.7	0.5	18.5	0.2	**1.4**	2.1	19.5
Segmenter, ViT-S/16																				
D&S (Ours, `WO`)	**74.1**	7.0	**65.7**	**6.6**	1.0	**24.9**	**4.3**	**16.6**	64.8	1.8	3.7	**45.9**	4.3	**57.3**	**1.7**	**19.9**	**1.3**	0.4	**12.1**	**21.8**

We observe that Drive&Segment outperforms the baseline PiCIE on 15 out of 19 classes. IoU gains (w.r.t. PiCIE trained on Waymo Open dataset) are significant for small-object classes such as *pole* (+23.2/+15.2 with Segmenter and ResNet18+FPN respectively), *traffic signs* (+15.4/+14.5), and *person* (+42.8/+36.9). They are also substantial for some classes that can cover larger image portions, *e.g.*, *road* (+15.6/+14.2), *vegetation* (+36.5/+38.5), *car* (+8.8/−3.8). The results of ResNet18+FPN are slightly worse on the *car* class because *car* instances are split into several pseudo-classes. Gains over *road* and *car* were expected since LiDAR data provide very good segments for these classes; it is more surprising to see gains on *vegetation*, a class that is not easily captured by LiDAR.

A.6 Unified cluster assignments across datasets.

In this experiment, we optimize the matching over a unified (joint) dataset, i.e., we compute the optimal pseudo label → ground-truth label assignment using Hungarian matching. The mIoU results in the Table 9 show the superiority of our approach over the previous state-of-the-art work [3] by outperforming it consistently across all datasets. The first column "*joint*" shows the mIoU computed using all samples from the datasets that we optimize the matching for.

Table 9. Per-dataset mIoU results with the Hungarian matching over a unified/joint dataset. The first column "*joint*" shows the mIoU computed using all samples from the datasets that we optimize the matching for.

model	joint	CS19	DZ	ND	ACDC
PiCIE	11.1	10.6	2.6	4.1	11.1
D&S (RN18)	15.5	16.4	7.2	11.2	12.7
D&S (Segm)	**19.4**	**21.4**	**10.8**	**14.3**	**12.9**

Table 10. Evaluation of learned features using k-NN pixel-wise classification. Results are produced by running k-NN with three different 100-image training sets [7] and computing the average (over the three runs) pixel accuracy on the Cityscapes validation split. Results are reported with the Pixel Accuracy (PA) metric.

method	$k=1$	$k=5$	$k=20$
supervised	76.9	79.4	81.2
PiCIE [3]	74.3 (-2.6)	78.0 (-1.4)	79.1 (-2.1)
Drive&Segment	81.1 (+4.2)	83.2 (+3.8)	84.7 (+3.5)

Fig. 8. Feature visualization. We do PCA analysis of the pixel-wise decoder features from each image (independently between the different images) and visualize the three first PCA components as an RGB image. 'Segm.' stands for Segmenter with ViT-S/16 and 'RN' for ResNet18+FPN.

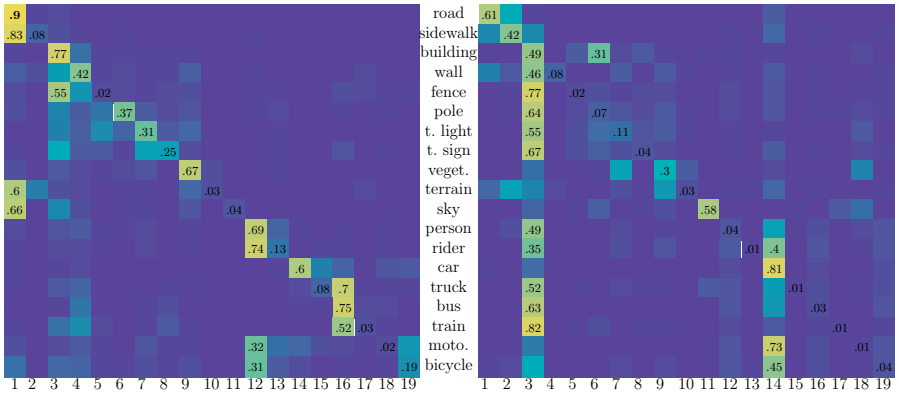
B Analyzing learned representations

B.1 k-NN evaluation of learned representations

To evaluate the quality of the learned representations, we compare the representations produced by a ResNet18 backbone trained (a) on Imagenet in a fully-supervised fashion for the classification task, (b) using PiCIE [3] trained on Waymo Open, and (c) using our Drive&Segment trained on Waymo Open. For this comparison we perform k-NN based pixel-wise classification on the Cityscapes validation set using a *low-shot scenario* where only 100 Cityscapes training images are available (we consider three random splits of 100 images from [7] and report the average results). Our goal is to analyze the ability of the representations to learn with a few training examples. In Table 10, we report results in terms of pixel accuracy for $k \in \{1, 5, 20\}$ and observe that Drive&Segment outperforms both the supervised baseline and PiCIE [3].

B.2 Representation analysis via PCA

In Figure 8, we visualize the three main PCA components of the *decoder* features as RGB. We observe that our features learned with Segmenter separate better object classes.



(a) Drive&Segment, Segmenter S (b) PiCIE [3]

Fig. 9. Row-normalized confusion matrices. Columns (pseudo-classes) are re-ordered based on the matching with rows (GT classes) from the Hungarian algorithm with resulting values on the diagonal. The higher the number, the better.

B.3 Confusion matrices for class mapping

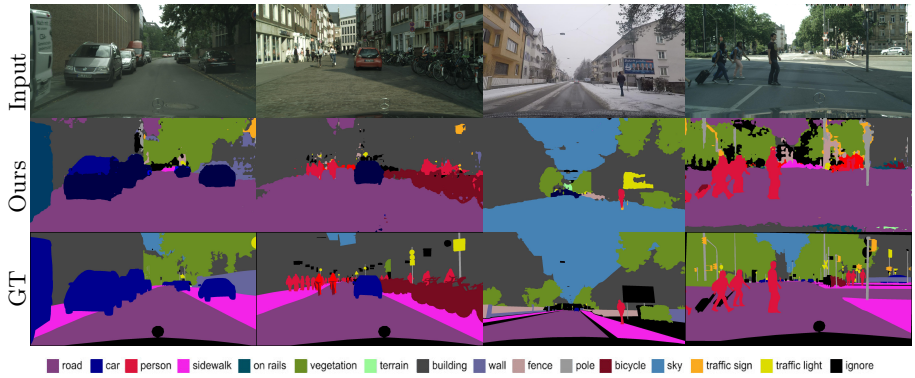
Here we analyze the confusion matrices, presented in Figure 9, which provide the mapping between ground truth and pseudo classes. For each confusion matrix, we reorder the columns based on the matching obtained from the Hungarian algorithm, and L_1-normalize the values per row, i.e., per ground-truth class (for simplicity, we do not illustrate the un-matched pseudo-classes in the figures). Thus, a value of 1 would signify that all pixels in a ground-truth class belong to a single pseudo-class. Moreover, due to the reordering, the largest values should ideally be on the diagonal of the confusion matrix.

For each row, the highest and the diagonal entry are reported. We note that, for Drive&Segment (Fig. 9(a)), 90% of the road pixels are covered by the first pseudo-class. However, this pseudo-class also covers large portions of sidewalk and vegetation as all these labels belong to ground pixels and hence are segmented together by our LiDAR-based segment proposal mechanism. Similarly, pseudo-class 12 overlaps person, rider, motorcycle and bicycle, i.e., with human-related ground-truth classes. Regarding PiCIE (Fig. 9(b)), only a few pseudo-classes have a significant overlap with ground-truth classes. In particular, the pseudo-class 3 overlaps with the majority of the ground-truth classes.

B.4 Failure cases

The main limitations of our Drive&Segment approach are discussed in Section 4.4 of the main paper. Here, we show some qualitative examples of these failure modes and discuss their roots.

The first limitation of Drive&Segment is the complete absence of pseudo-labeled training data for the *sky* class. This is because the LiDAR data do not



(a) mixed labels (b) good shape, wrong class (c) sky ↔ road confusion

Fig. 10. Failure cases. **(a)** Due to the noise in the training data (discussed in Section B.4; images and LiDAR point clouds come from the Waymo Open [11] dataset), Drive&Segment sometimes predicts multiple pseudo-labels inside the same object (here different shades of blue inside the car on the left). **(b)** Objects that belong to the same semantic category (*e.g.*, *cars*) might end-up clustered into different pseudo-classes due to differences in appearance (*e.g.*, a separate pseudo-class that corresponds to the rear of the cars). **(c)** The *road*↔*sky* misplacement/confusion is caused by the absence of sky-occupied labeled pixels at training as they are not covered by the LiDAR data. Therefore, the model assigns the most common label to the sky, which is the pseudo-label that corresponds to the road. This leads to either predicting the road as *sky* (third column), or predicting sky as *road* (fourth column), depending on the outcome of the Hungarian matching.

capture the sky. As a consequence, our models learn to classify the *sky* pixels as *road* (see the "sky" row of the confusion matrix in Figure 9a), which is the most dominant (pseudo-)label in the data. We provide examples of this behavior in Figure 10(c).

The second most common failure mode is inherited from the object proposal method that relies only on geometry-derived features. Specifically, the segment proposal method might over-segment an object, as visible in Figure 10(a). As a consequence, our models might learn to make predictions that mix multiple pseudo-labels in one object.

Third, we face the issue of class over clustering. This means that certain ground-truth classes, such as cars, are not contained just in a single cluster, but are in multiple of them. This is a problem during the evaluation using Hungarian matching. It results in some high off-diagonal values in the confusion matrix as only one pseudo label can be assigned to each GT class. An example of this behavior is shown in Figures 10(b) and 11.

Finally, our LiDAR-based proposal method groups all points from the ground plane into a single segment, without being able to distinguish the various ground-plane classes (*e.g.*, *road*, *sidewalk* and *terrain*) that are defined in the image

Fig. 11. Failure case. Multiple pseudo classes (**A**,**B**,**C**) for the same GT class "car".

domain. Figure 10 provides examples of this failure mode. This phenomenon is also well visible in Figure 9a.

C Additional qualitative results

C.1 Qualitative comparison to previous work

We show a qualitative comparison with IIC [8] and PiCIE [3] in Figure 12. For a fair comparison, we use the same samples and the visualization protocol as in [3]. Note that these samples come from the PiCIE and IIC training set, namely from the *train* set of the Cityscapes [4] dataset, while for our method these are only test samples. In Figure 12, note how our Drive&Segment is able to segment the *person* class, while neither IIC nor PiCIE are capable to do so.

C.2 Qualitative Results

In Figures 13 and 14, we report Drive&Segment predictions on Cityscapes validation images. In spite of the domain gap between the training dataset (Waymo Open Dataset [11] with images from US cities) and the Cityscapes test set, our approach produces convincing results. Furthermore, in Figure 15, we report qualitative results of our method pretrained on *daytime-only* images and evaluated on *out-of-training-distribution* splits of ACDC [10], *e.g.*, *night*, *snow* or *fog*. We discuss the main failure modes in Section B.4.

D Evaluating Drive&Segment with supervised fine-tuning

The goal of our work is to train image segmentation models without any human annotation. Here, we evaluate with some preliminary experiments the applicability of the proposed Drive&Segment method on a different but related task, that of *self-supervised pre-training* of semantic segmentation networks (i.e., self-supervised feature learning). Specifically, we take the ResNet18+FPN model trained with Drive&Segment, replace its last linear prediction layer with a new layer that has as many outputs as classes in Cityscapes (19), and fine-tune the resulting network on the Cityscapes [4] dataset using available human annotations. We compare against (a) using PiCIE [3] for self-supervised pre-training and (b) *supervised* pre-training on ImageNet [6].

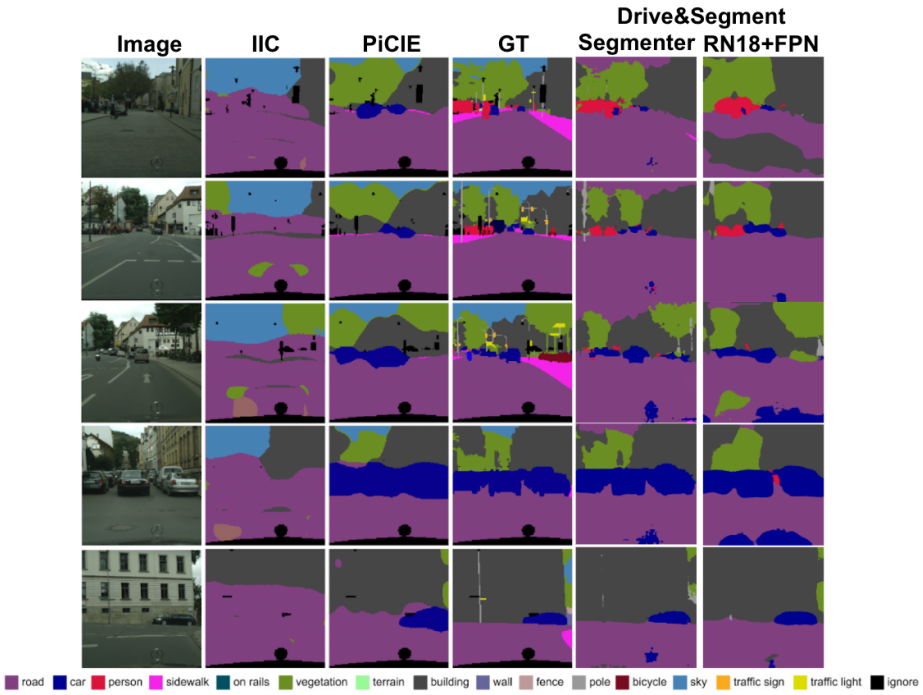


Fig. 12. Qualitative comparison of PiCIE [3], IIC [8] and our Drive&Segment approach on PiCIE *training* samples. For a fair comparison, we use the same visualization procedure as in [3]. Results are shown on center-cropped Cityscapes training images. Note that our method is able to capture objects' contours much better and to segment categories such as *person* that are not visible in IIC or PiCIE results.

Table 11. Supervised fine-tuning on the Cityscapes [4] semantic segmentation task. Results report mean Intersetion over Union (mIoU). We fine-tune the pre-trained ResNet18+FPN networks on either the entire Cityscapes training split ('Full Cityscapes') or only 100 images from the training split ('Low-shot') [7] and test on the Cityscapes validation split. 'Linear' fine-tunes only the last linear layer, 'Decoder+Linear' fine-tunes the FPN decoder and the last linear layer, and 'End-to-End' fine-tunes the entire network.

	Full Cityscapes		Low-shot
Pre-training	Linear	Decoder+Linear	End-to-End
PiCIE [3]	17.4	29.5	30.4
Imagenet (supervised)	25.7	41.9	48.2
Drive&Segment	**36.4**	**46.4**	**49.2**

We evaluate the different pre-training approaches with three fine-tuning setups. The first setup is to freeze both the ResNet18 backbone and the FPN decoder (i.e., keep their pre-trained weights fixed) and fine-tune only the last

linear prediction layer. The second setup is to freeze only the ResNet18 backbone and fine-tune both the FPN decoder and the last linear layer. In both cases, we train on the entire training split (2975 images) of Cityscapes. The goal of these first two setups is to evaluate the quality of the pre-trained ResNet18+FPN (1st setup) or ResNet18 (2nd setup) features as they are. The third setup targets the *low-shot scenario*: the segmentation network is fine-tuned end-to-end using only 100 Cityscapes training images (we consider three random splits of 100 images from [7]). The purpose of this setup is to evaluate the strength of the pre-trained network in a regime where only a few annotations are available for fine-tuning.

In the first two setups we train for $40k$ iterations and we train for $4k$ in the low-shot setup. In all setups, we use SGD with momentum set to 0.9, weight decay to 0.0005, and mini-batches of size 8. During training, we use random image scaling (by a ratio in $[0.5, 2.0]$), random cropping (with size 769), and horizontal flipping. At test time, we use the original image size and horizontal flip augmentations. The learning rates were tuned for each fine-tuning setup and each evaluated method separately.

We report results in Table 11. Although our method was not designed or optimized for self-supervised feature pre-training, it still provides promising results that surpass both PiCIE and ImageNet pre-training.

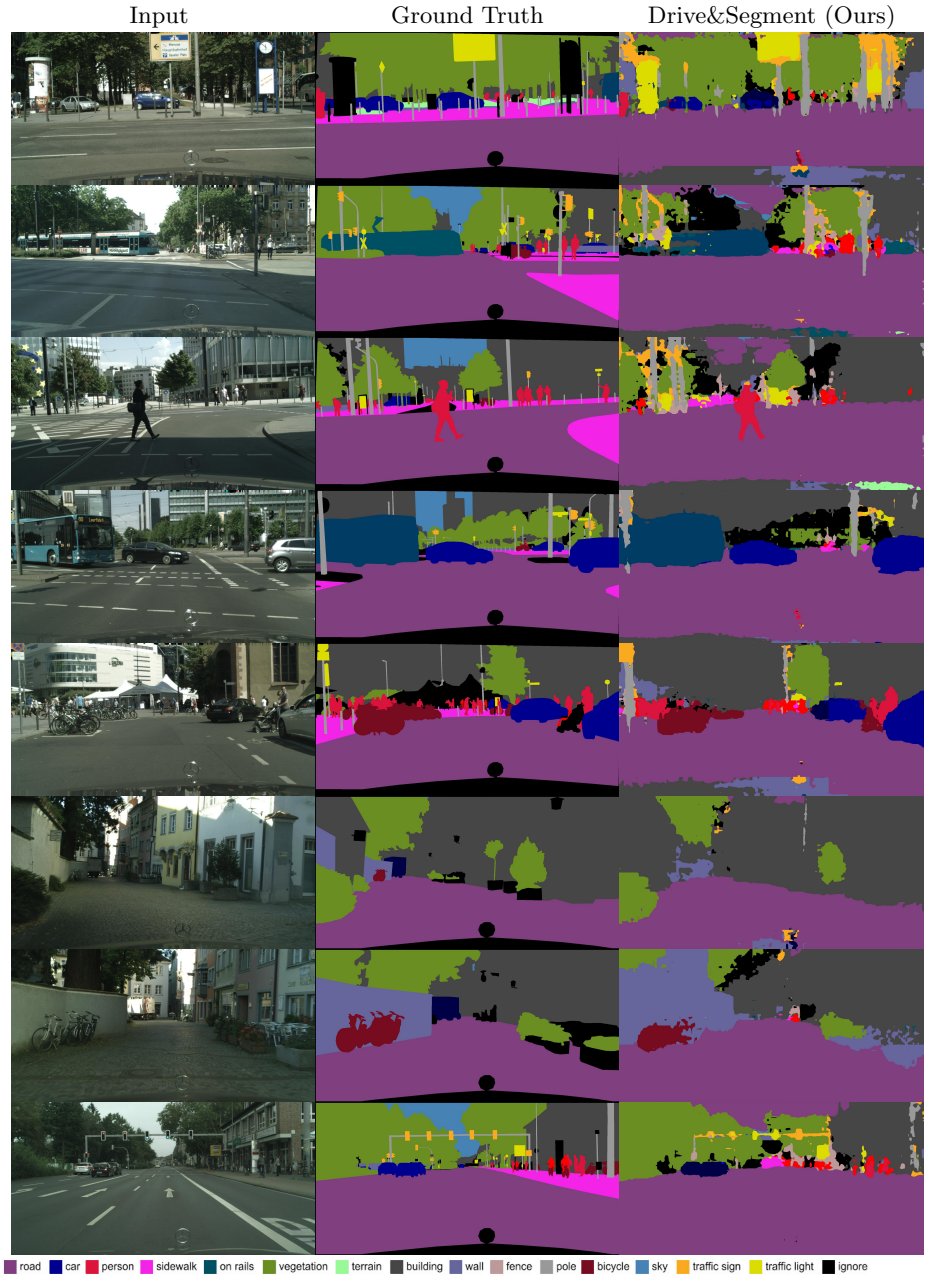


Fig. 13. Qualitative results for unsupervised semantic segmentation using our Drive&Segment approach on the validation split of the Cityscapes dataset. The matching between our pseudo-classes and the set of ground-truth classes is obtained using the Hungarian algorithm.

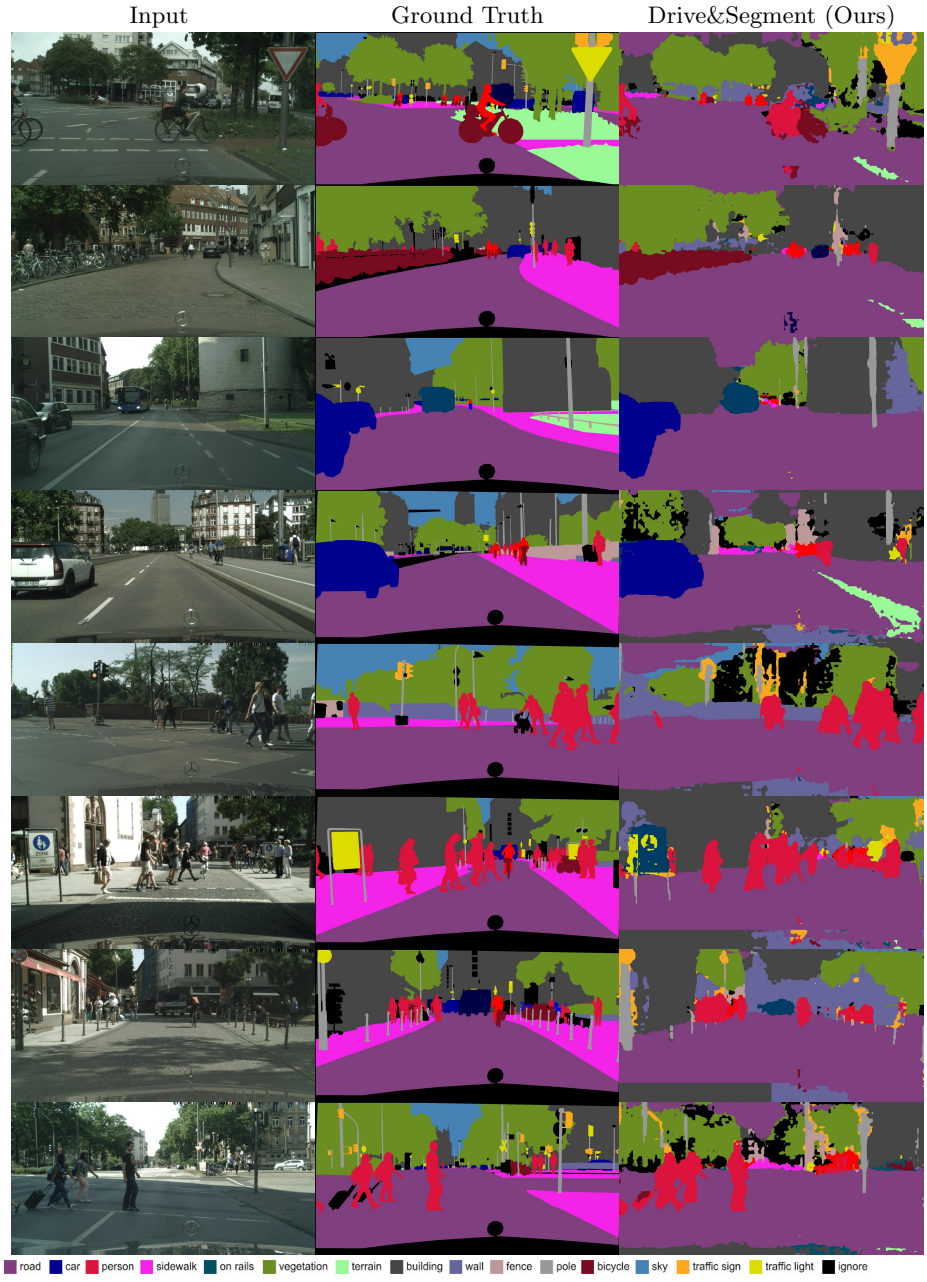
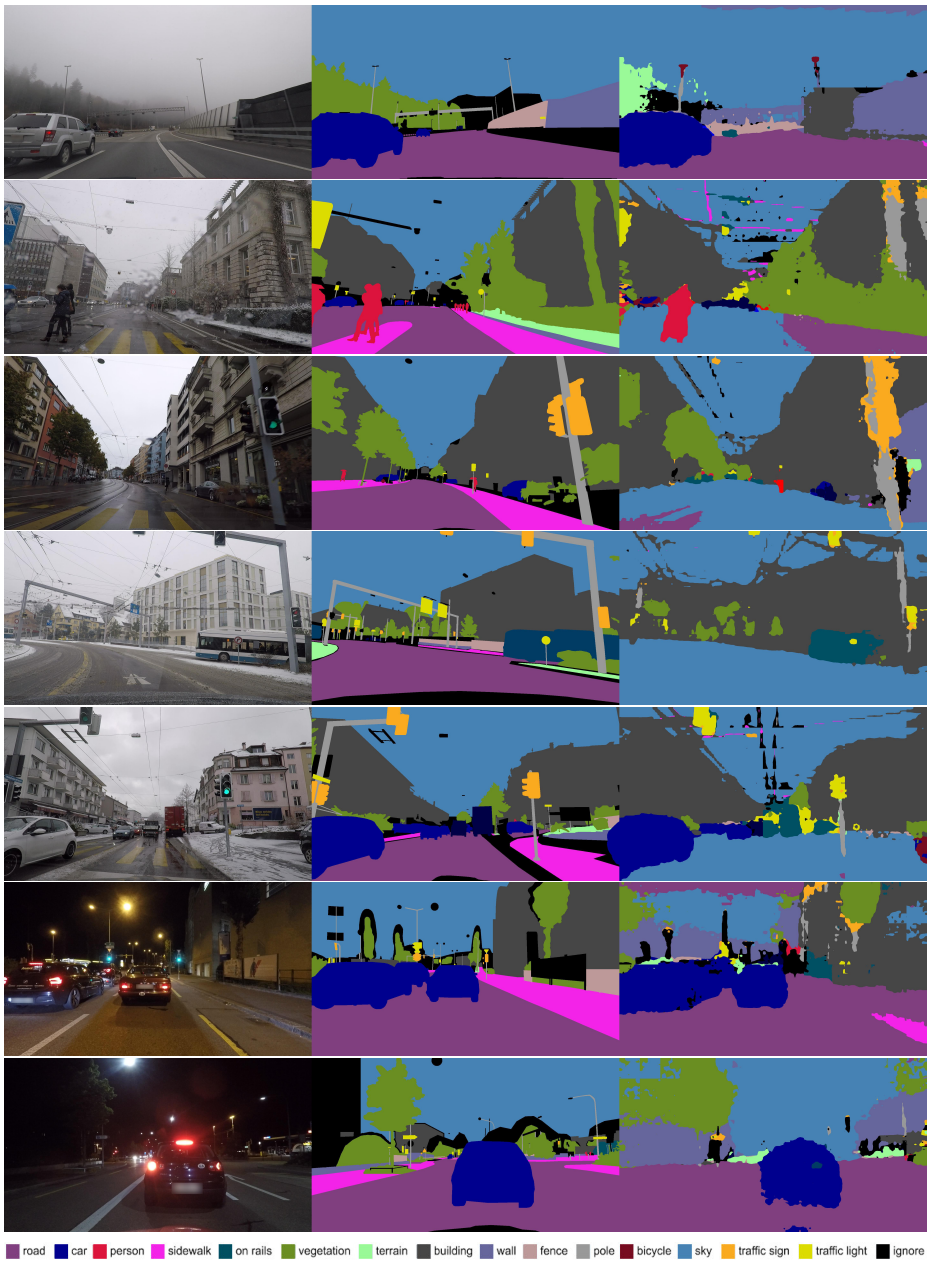


Fig. 14. Qualitative results for unsupervised semantic segmentation using our Drive&Segment approach on the validation split of the Cityscapes dataset. The matching between our pseudo-classes and the set of ground-truth classes is obtained using the Hungarian algorithm.



Input Ground Truth Drive&Segment (Ours)

Fig. 15. Waymo Open Dataset *day* → ACDC [10] {*fog, rain, snow, night*}. Qualitative results of our Drive&Segment model trained on the daytime images from the Waymo Open Dataset and used to segment samples from the ACDC [10] dataset with various adverse conditions. In rows 2-5 the *ground* is incorrectly segmented as *sky*. This failure mode is further discussed in Section B.4.

References

1. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: CVPR (2020) 1, 2
2. Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. In: ECCV (2018) 2, 3
3. Cho, J.H., Mall, U., Bala, K., Hariharan, B.: PiCIE: Unsupervised semantic segmentation using invariance and equivariance in clustering. In: CVPR (2021) 2, 3, 4, 5, 6, 8, 9
4. 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) 2, 3, 4, 8, 9
5. Dai, D., Van Gool, L.: Dark model adaptation: Semantic image segmentation from daytime to nighttime. In: IEEE ITSC (2018) 2, 3
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR (2009) 8
7. French, G., Laine, S., Aila, T., Mackiewicz, M., Finlayson, G.: Semi-supervised semantic segmentation needs strong, varied perturbations. BMVC (2020) 5, 9, 10
8. Ji, X., Henriques, J.F., Vedaldi, A.: Invariant information clustering for unsupervised image classification and segmentation. In: ICCV (2019) 3, 8, 9
9. Sakaridis, C., Dai, D., Van Gool, L.: Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. IEEE TPAMI (2020) 2, 3
10. Sakaridis, C., Dai, D., Van Gool, L.: ACDC: The adverse conditions dataset with correspondences for semantic driving scene understanding. In: ICCV (2021) 2, 3, 8, 13
11. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: CVPR (2020) 4, 7, 8