Traffic Scene Parsing through the TSP6K Dataset

Peng-Tao Jiang[1,2*] Yuqi Yang[1*] Yang Cao[3] Qibin Hou[1,4†] Ming-Ming Cheng[1,4] Chunhua Shen[2]
[1]VCIP, CS, Nankai University [2]Zhejiang University [3]HKUST [4]NKIARI, Shenzhen Futian
pt.jiang@mail.nankai.edu.cn, yangyq2000@mail.nankai.edu.cn, andrewhoux@gmail.com

Abstract

Traffic scene perception in computer vision is a critically important task to achieve intelligent cities. To date, most existing datasets focus on autonomous driving scenes. We observe that the models trained on those driving datasets often yield unsatisfactory results on traffic monitoring scenes. However, little effort has been put into improving the traffic monitoring scene understanding, mainly due to the lack of specific datasets. To fill this gap, we introduce a specialized traffic monitoring dataset, termed TSP6K, containing images from the traffic monitoring scenario, with high-quality pixel-level and instance-level annotations. The TSP6K dataset captures more crowded traffic scenes with several times more traffic participants than the existing driving scenes. We perform a detailed analysis of the dataset and comprehensively evaluate previous popular scene parsing methods, instance segmentation methods and unsupervised domain adaption methods. Furthermore, considering the vast difference in instance sizes, we propose a detail refining decoder for scene parsing, which recovers the details of different semantic regions in traffic scenes owing to the proposed TSP6K dataset. Experiments show its effectiveness in parsing the traffic monitoring scenes. Code and dataset are available at https://github.com/PengtaoJiang/TSP6K.

1. Introduction

As a classic and important computer vision task, the scene parsing task aims to segment the semantic objects and stuff from the given images. Nowadays, with the emergence of large-scale scene understanding datasets, such as ADE20K [101] and COCO-Stuff [5], has greatly promoted the development of scene understanding algorithms [21, 43, 52, 57, 83, 98, 100]. Many application scenarios, such as robot navigation [18, 42] and medical diagnosis [62], benefit from the advanced scene understanding algorithms. As an important case of scene understanding, traffic scene understanding focuses on understanding urban street scenes, where the most frequently appeared instances are humans, vehicles, and traffic signs. To date, there are already many large-scale publicly available traffic scene datasets, such as KITTI [25], Cityscapes [17], and BDD100K [87]. Benefiting from these finely anontated datasets, the segmentation performance of the recent scene understanding approaches [13, 30, 49, 51, 66, 78, 94, 102] has also been considerably improved.

A characteristic of these traffic datasets is that they are mostly collected from a driving platform, such as a driving car, and hence are more suitable for the autonomous driving scenario. However, little attention has been spent on the traffic monitoring scenes. Traffic monitoring scenes are usually captured by the shooting platform high-hanging (4.5-6 meters) in the street, which offers a rich vein of information on traffic flow [44, 58]. The high-hanging shooting platform usually observes more traffic participants than the driving ones, especially at the crossing. We observe that deep learning models trained on these existing traffic datasets can obtain poor results in parsing traffic monitoring scenes, possibly because of the domain gap. Although analyzing the traffic monitoring scenes is in demand for many applications, such as traffic flow analysis [44, 58], no current traffic scene datasets are available for facilitating such research, to the best of our knowledge.

To facilitate the research on parsing the traffic monitoring scenes, we construct a specific dataset for traffic scene analysis and present it in this paper. Specifically, we carefully collect many traffic images from the urban road shooting platforms at different locations. To keep the diversity of our dataset, we collect images from hundreds of traffic scenes at different times of the day. To conduct semantic segmentation and instance segmentation on this dataset, we ask annotators to finely annotate them with high-quality semantic and instance-level labels. Due to the expensive labor for annotations, we finally obtained 6,000 finely anontated traffic images. In Fig. 1, we have shown some traffic images and their corresponding semantic-level and instance-level labels. Using these finely anontated labels, we perform a compre-

*The first two authors contributed equally to this work. Part of this work was done when P.-T. Jiang was a postdoc researcher at Zhejiang University.
†Q. Hou is the corresponding author.

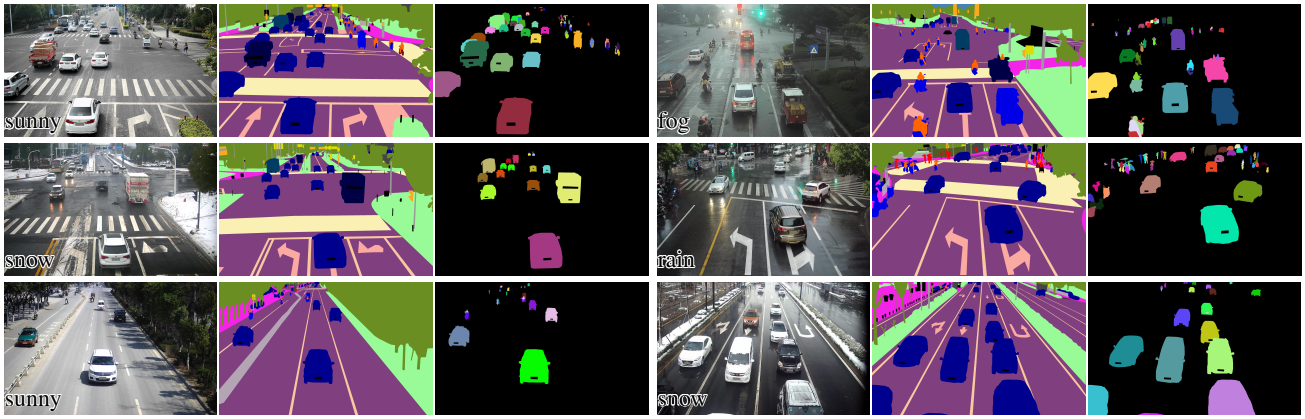


Figure 1. Examples are randomly picked from the TSP6K dataset. Each image is associated with its corresponding semantic label and instance label. We have masked the vehicle plates for privacy protection.

hensive study about the traffic monitoring scenes. The characteristics of this dataset are summarized as follows: **1)** the largest traffic monitoring datasets, **2)** much more crowded scenes, **3)** wide variance of instance sizes, and **4)** a large domain gap between the driving scenes and the monitoring scenes. Based on the proposed TSP6K dataset, we evaluate a few classic scene parsing methods, instance segmentation methods, and unsupervised domain adaption methods. We show and analyze the performance of different methods on the proposed TSP6K dataset.

In addition, we propose a detail refining decoder for image segmentation on TSP6K. The detail refining decoder utilizes the encoder-decoder structure and refines the high-resolution features with a region refining module. The region refining module utilizes the attention mechanism and computes the attention between the pixels and each region token. The attention is further used to refine the pixel relationships in different semantic regions. Taking the backbone of the SegNeXt [27] as our encoder, our proposed detail refining decoder method achieves 75.8% mIoU score and 58.4% iIoU score on the TSP6K validation set, which are 1.2% and 1.1% higher than those of the state-of-the-art SegNeXt work. To summarize, our main contributions are as follows:

- We propose a specialized traffic dataset for researching the task of traffic monitoring scene parsing, termed TSP6K, which collects images spanning various scenes from the urban road shooting platform. We also provide pixel-level annotations of semantic labels and instance labels.
- Based on the TSP6K dataset, we evaluate the performance of previous scene parsing methods and instance segmentation methods on traffic monitoring scenes. Moreover, the TSP6K dataset can also serve as an additional supplement for evaluating unsupervised domain adaption methods for traffic monitoring applications.
- To improve traffic monitoring scene parsing, we propose a detail refining decoder that learns region tokens to refine different regions on high-resolution features. Experiments validate the effectiveness of the proposed decoder.

2. Related Work

2.1. Scene Parsing Datasets

Scene parsing datasets with full pixel-wise annotations are utilized for training and evaluating the scene parsing algorithms. As an early one, the PASCAL VOC dataset [22] was proposed in a challenge that aims to parse the objects of 20 carefully selected classes in each image. Later, the community proposed more complex datasets with many more classes, such as COCO [54] and ADE20K [101]. The scenes in the above datasets span a wide range. Different from these datasets, there are also some datasets focusing on particular scenes, such as the traffic scenes. There exist many traffic scene parsing datasets [19, 39, 60, 64, 72, 90, 91], such as KITTI [25], Cityscapes [17], ACDC [65], and BDD100K [87]. These traffic-parsing datasets annotate the most frequent classes in the traffic scenes, such as the traffic sign, rider, and vehicles, *etc*. Based on these finely annotated traffic datasets, the approaches based on the neural networks have achieved great success in parsing traffic scenes.

Despite the success of the above datasets, we find that the traffic scenes in these datasets all from the driving platform. Models trained on these datasets often behave not well on the traffic monitoring scenes, which play an important role in traffic flow analysis. In addition, the monitoring scenes usually capture much more traffic participants than the driving scenes. The proposed TSP6K dataset is different from the driving datasets, aiming at improving the performance of scene parsing models on monitoring scenes, which can be regarded as a supplement to the current traffic datasets. Furthermore, Kirillov *et al*. [46] has proposed a large segmentation dataset, SA-1B, containing 1 billion masks. SA-1B also contains some monitoring traffic images. However, the segmentation masks in SA-1B are all class-agnostic. In

contrast to that, the segmentation masks in our dataset are all class-known.

2.2. Scene Parsing Approaches

Convolutional neural networks have greatly facilitated the development of scene parsing approaches. Typically, Long *et al*. [57] first proposed a fully convolutional network (FCN) that generates dense predictions for scene parsing. Later, some approaches, such as the popular DeepLab [9, 10] and PSPNet [98], benefit from large receptive fields and multi-scale features, improving the performance by a large margin. Besides, there are also some approaches [3, 12, 13, 52], utilizing the encoder-decoder structure to refine the low-resolution coarse predictions with the details of high-resolution features. Except for the simple convolutions, researchers [36, 41, 89, 99] found that the attention mechanism [29] can significantly improve scene parsing networks due to their ability to model long-range dependencies. In addition, there are also some works [79, 85, 86, 95, 97] exploring real-time scene parsing algorithms, which take advantage of self-attention in efficient ways.

Recently, with the successful introduction of Transformers into image recognition [20], researchers have attempted to apply Transformers to the segmentation task [14, 15, 66, 83, 100]. Interestingly, some recent works [27, 28, 35] show that convolutional neural networks can perform better than Transformer-based models on the scene parsing task. In our dataset, we also observe similar results. SegNeXt [27] achieves the best performance on our TSP6K dataset using even fewer parameters than other works. The proposed method in this work also adopts the backbone of the SegNeXt work. But different from it, we design a detail refining decoder, which is more suitable for processing high-resolution images than the one used in SegNeXt.

2.3. Instance Segmentation Approaches

Instance segmentation aims to segment and distinguish each instance of the same class. Previous instance segmentation approaches can be roughly divided into two groups, box-dependent methods [4, 7, 23, 31, 40, 55], and box-free methods [68, 75, 76, 82]. Box-dependent methods first detect the bounding box of the target object, and then perform binary segmentation in the box region. In contrast, box-free methods directly generate the instance mask for each instance and classify it in parallel. In this paper, we select several methods for each category and evaluate them on TSP6K.

2.4. Unsupervised Domain Adaption Approaches

Unsupervised domain adaption (UDA) aims to adapt the models trained on one domain (with segmentation labels) to a new domain (without segmentation labels). In recent years, lots of UDA approaches [33, 70, 73, 84] for scene parsing emerge to address the domain discrepancy. The

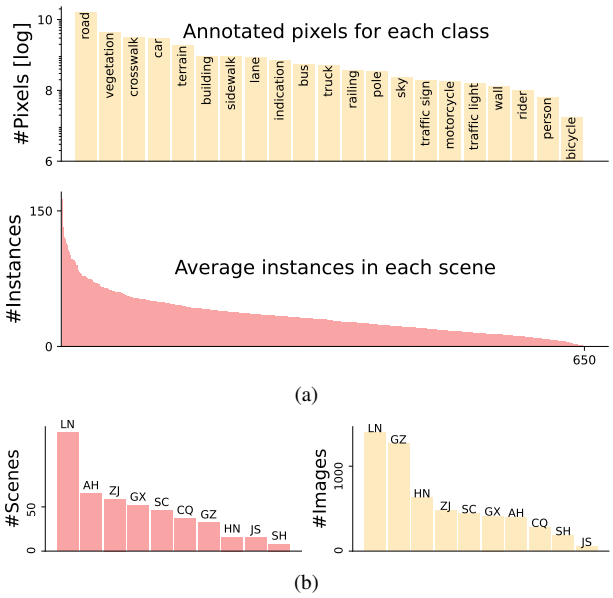


Figure 2. (a) Class and scene information of the TSP6K dataset. (b) The geographic distribution of the scene and image.

UDA approaches mainly fall into two categories: adversarial training-based methods [33, 34, 70, 71, 73, 74], and self-training-based methods [47, 50, 59, 77, 93, 103, 104]. The adversarial training-based methods attempt to align the feature representations or network predictions of the source and target domains. The self-training-based methods generate pseudo masks for the target domain to train segmentation networks. Previous UDA methods are usually evaluated by adapting from the synthetic traffic datasets (GTA5 [61] or SYNTHIA [63]) to the real traffic datasets (Cityscapes [17]). In this paper, we evaluate the UDA methods for adapting both the synthetic and real driving scenes (SYNTHIA [63]), Cityscapes [17]) to the monitoring scenes (TSP6K).

3. Dataset and Analysis

In this section, we introduce the details for constructing the monitoring dataset and perform a comprehensive analysis of the proposed TSP6K dataset.

3.1. Data Collection

One significant aspect of researching the traffic monitoring scenes is data. Once we construct a dataset for the monitoring scenes, the community researchers can improve the scene parsing results based on the novel data characteristics. To facilitate the research, we aim to build a dataset specifically for researching the traffic monitoring scenes by collecting a large number of images from the high-hanging shooting platforms on different streets. To ensure data diversity, the collection locations and weather conditions are highly considered. Specifically, we collect the traffic images

Table 1. Comparisons among different traffic scene parsing datasets. **Avg TP** denotes the number of the average traffic participants in each image. **TP > 50** denotes the number of images that contain more than 50 traffic participants. As the instance labels of the test sets in other datasets are not available, we all count the traffic participants in the training and validation sets. We can see that our TSP6K dataset contains more traffic images with more than 50 traffic participants when compared with other datasets.

Type	Datasets	Class	Weather	#Images	Resolution	Annotation	Avg TP	TP > 50	TP > 75	TP > 100
Driving	KITTI [25]	19	Good	7,481	1,241×375	Pixel&Inst	4.9	0	0	0
	Cityscapes [17]	19	Good	5,000	2,048×1,024	Pixel&Inst	18.8	54	10	4
	WildDash 2 [90]	26	Diverse	5,068	1920×1080	Pixel&Inst	9.0	12	4	2
	Mapillary [60]	65	Diverse	25,000	3,436×2,486	Pixel&Inst	12.3	102	15	3
	ACDC [65]	19	Diverse	3,142	1080×1920	Pixel&Inst	6.3	0	0	0
	BDD100K [87]	40	Good	10,000	1,280×720	Pixel&Inst	12.8	5	0	0
Monitoring	UrbanTracker [45]	7	Good	5(videos)	1,035×632	Box	3.7	0	0	0
	CityFlow [67]	1	Good	40(videos)	540×960	Box	2.0	-	-	-
	AAU RainSnow [1]	3	Diverse	22(videos)	640×480	Pixel	6.6	0	0	0
	TSP6K (ours)	21	Diverse	6,000	2,942×1,989	Pixel&Inst	42.0	1,227	367	73

Table 2. Statistics of traffic participants in traffic images. '#H.' and '#V.' denote #Humans and #Vehicles, respectively.

Datasets	#Humans $[10^3]$	#Vehicles $[10^3]$	#H./images	#V./images
KITTI [25]	6.1	30.3	0.8	4.1
Cityscapes [17]	24.4	41.0	7.0	11.8
Mapillary [60]	6.7	17.8	3.4	8.9
Wilddash2 [90]	11.6	26.8	2.7	6.3
ACDC [65]	3.8	15.9	1.2	5.1
BDD100K [87]	11.7	90.3	1.5	11.3
TSP6K (ours)	64.0	188.2	10.7	31.3

from about 10 Chinese provinces with more than 600 scenes. In Fig. 2(b), we have shown the geographic distribution of scenes and images. As the crossing and pedestrian crossing are an essential part of traffic scenes, where congestion and accidents often occur, we keep a majority of the traffic scenes containing the crossing. Besides, considering the weather diversity, we select the traffic images under various weather conditions, including sunny and cloudy day, rain, fog, and snow. As a result, we finally selected 6,000 traffic images.

3.2. Data Annotation

After collecting data, we start to annotate the traffic images. The complete annotated classes are shown in Fig. 2(a). Specifically, we annotate 21 classes, where most of the classes are the same as the class definition in Cityscapes [17]. We remove the unseen class 'train' in our dataset and add three new classes. As the indications on the road are vital for understanding the monitoring scenes, we ask the annotators to label three indication classes for traffic, namely crosswalks, driving indications, and lanes. Besides, we have annotated the instance mask for each traffic participant.

Similar to the annotation policy of Cityscapes [17], the traffic images are also annotated from back to front. To keep the quality of the labels, we design a double-check mechanism. Specifically, the images are split into 30 groups, each of which contains 200 images. When the annotators finish labeling the images, we pick 30% of 200 images and check if there exist class labeling errors. If there exist class labeling errors in the selected images, we ask the annotator to check all the images in this group until there are no class labeling errors.

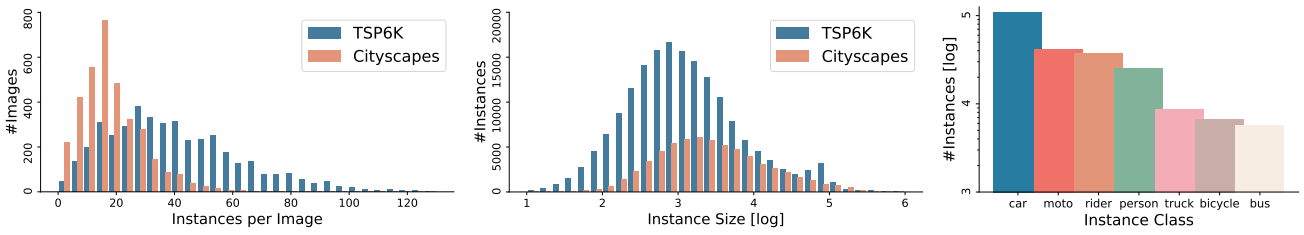
3.3. Data Split

The dataset is divided into three splits for training, validation, and test according to the ratio of 5:2:3. Images collected from different scenes are randomly split into different sets. In total, there are 2,999, 1,207, and 1,794 images for the training, validation, and test sets, respectively.

3.4. Data Analysis

We compare the TSP6K dataset with previous traffic datasets regarding the scene type, instance density, scale variance of instances, and domain gaps. In Tab. 1 and Tab. 2, we have listed the comparison among different traffic datasets. The characteristics of TSP6K can be summarized as follows:

Largest traffic monitoring dataset: To the best of our knowledge, most previous popular traffic datasets focus on the driving scenes. The images of these datasets are collected from the driving platform. There are also several datasets [45, 67] including the traffic monitoring scenes, shown in Tab. 1. However, they mainly focus on traffic participant tracking and only provide class-agnostic bounding box annotations. Different from them, we address traffic monitoring scene parsing and provide semantic and instance annotations. Compared with the existing traffic monitoring datasets, TSP6K contains much more labeled semantic classes, available instance segmentation, larger image resolution and a



(a) (b) (c)

Figure 3. Data analysis of the TSP6K dataset. (a) The distribution of the number of instances in each image. (b) The distribution of the instance sizes. (c) The number of instances for each category.

much larger number of images.

Much more crowded scenes: One of the most important characteristics is that the TSP6K dataset contains more crowded images than those in driving datasets. Since the majority of the traffic scenes are shooted at the crossing, the instance density on the road is much larger than the driving scenes. In Tab. 1, it can be seen that the driving datasets have few images containing more than 50 instances. In contrast, our TSP6K dataset has a large number of images containing more than 50 instances, occupying about 30% of the images in the training and validation sets. Moreover, as shown in Tab. 2, there are 10.7 humans and 31.3 vehicles on average in TSP6K, which exceeds other driving datasets several times. In addition, it can be seen that the existing monitoring datasets contain fewer annotated instances than driving datasets. This can be mainly attributed to the incomplete annotation where only the moving vehicles are annotated.

Wide variance in the instance sizes: For the monitoring scenes, the scale difference of the instances in the front and end is very large. As shown in Fig. 3(b), the instance size of TSP6K spans a wider range than Cityscapes. Furthermore, TSP6K also contains more small traffic instances than Cityscapes. The high-hanging platform usually has a much broader view than the driving platform. Thus, it can capture much more content in the distance. The huge variance of the instance sizes shows real traffic scenarios.

Large domain gap: There exists a large domain gap between TSP6K and Cityscapes/BDD100K. The models trained on driving datasets usually achieve low-quality results on monitoring scenes. Furthermore, for UDA from SYNTHIA to Cityscapes, HRDA [38] achieves a 65.8% mIoU score. However, HRDA only achieves a 45.4% mIoU score, which also indicates the large domain gap between the driving scenes and the monitoring scenes. Providing a high-quality human-labeled dataset for analyzing the effectiveness of different methods of monitoring scenes will be beneficial for the community. It enables the researchers to cross-validate the effectiveness of the segmentation methods, instance segmentation methods, and unsupervised segmentation methods on the traffic monitoring scenes.

4. Evaluating Segmentation Methods

4.1. Implementation Details

We run all the scene parsing methods based on a popular codebase, mmsegmentation [16]. All the models are trained on a node with 8 NVIDIA A100 GPUs. For CNN-based methods, the input size is set to 769×769. For transformer-based methods, the input size is set to 1024×1024. We train all the methods for 160,000 iterations with a batch size of 16. As SETR [100] consumes large GPU memories, the input size is set to 769×769, and the batch size is set to 8. Furthermore, we utilize the default data augmentations in mmsegmentation [16]. We utilize the mIoU [57] metric to evaluate the performance of the scene parsing methods. As mentioned in [17], the mIoU metric is biased to the object instances with large sizes. However, the monitoring traffic scene is full of small traffic participants. To better evaluate the instances of the traffic participants, we utilize the iIoU metric over all classes containing instances, following [17].

4.2. Performance Analysis

The evaluating results of different methods can be found in Tab. 3. The scene parsing methods can be roughly divided into several groups. In the following, we mainly discuss the methods using encoder-decoder structure, the self-attention mechanism and the transformer structure.

Encoder-decoder structure: The methods based on the encoder-decoder structure utilize the high-resolution low-level features to refine the details of segmentation maps. UperNet [80] and DeepLabv3+ [12] apply the encoder-decoder structure to the segmentation network. Compared with DeepLabv3 [11], DeepLabv3+ [12] utilizes the high-resolution features can further improve the segmentation results by more than 0.6% mIoU score and 1% iIoU score on both two sets. We observe that the encoder-decoder structure is very useful for small object segmentation, where the iIoU score is improved by a large margin.

Self-attention mechanism is widely used in scene parsing methods. Among the evaluated methods, EncNet [92], DANet [24], EMANet [48], and CCNet [41] all utilize different kinds of self-attention mechanisms. Most of them obtain

Table 3. Evaluation results of previous scene parsing approaches on the TSP6K validation and test sets.

Methods	Publication	Backbone	Parameters	GFlops	Validation		Test	
					mIoU (%)	iIoU (%)	mIoU (%)	iIoU (%)
FCN [57]	CVPR'15	R50	49.5M	454.1	71.5	55.2	72.5	55.1
PSPNet [98]	CVPR'16	R50	49.0M	409.8	71.7	54.8	72.6	54.8
DeepLabv3 [11]	ArXiv'17	R50	68.1M	619.3	72.4	55.0	73.3	55.0
UperNet [80]	ECCV'18	R50	66.4M	541.0	72.4	55.2	73.1	55.0
DeepLabv3+ [12]	ECCV'18	R50	43.6M	404.8	73.1	56.1	73.9	56.3
PSANet [99]	ECCV'18	R50	59.1M	459.2	71.3	54.5	72.6	54.8
EMANet [48]	ICCV'19	R50	42.1M	386.8	72.0	55.5	72.9	55.5
EncNet [92]	CVPR'18	R50	35.9M	323.3	71.4	54.8	72.7	55.0
DANet [24]	CVPR'19	R50	49.9M	457.3	72.3	56.0	73.1	56.1
CCNet [41]	ICCV'19	R50	49.8M	460.2	72.0	55.3	73.1	55.3
KNet-UperNet [96]	NeurIPS'21	R50	62.2M	417.4	72.6	56.8	73.7	56.5
OCRNet [88]	ECCV'20	HR-w18	12.1M	215.3	73.2	55.3	73.7	55.1
SETR [100]	CVPR'21	ViT-Large	310.7M	478.3	70.5	44.9	70.7	45.0
SegFormer [83]	NeurIPS'21	MIT-B2	24.7M	72.0	72.9	54.6	73.8	54.9
SegFormer [83]	NeurIPS'21	MIT-B5	82.0M	120.8	74.5	56.7	74.8	56.7
Swin-UperNet [56]	ICCV'21	Swin-Base	121.3M	1184.6	74.9	57.4	75.6	57.2
SegNeXt [27]	NeurIPS'22	MSCAN-Base	27.6M	80.2	74.6	57.3	75.4	57.2
SegNeXt [27]	NeurIPS'22	MSCAN-Large	48.9M	258.6	74.8	57.7	75.6	57.6
DRD (Ours)	–	**MSCAN-Base**	**46.1M**	**90.1**	**75.8**	**58.4**	**75.9**	**58.0**

superior performance than FCN. Benefiting from the ability to model the long-range pixel dependence, the methods based on the self-attention mechanism can refine the final segmentation results and improve the performance. Among them, we observe that EncNet does not have performance gain compared to FCN. We analyze that EncNet utilizes the channel-wise self-attention mechanism to build the global context, which cannot preserve the local details well, especially for traffic monitoring scenes that contain different sizes of traffic participants.

Transformer structure has been successfully applied to the computer vision tasks [6, 20], which often achieves better recognition results than the convolutional neural network structures. SETR [100], Segformer [83], Swin-UperNet [56], and SegNeXt [27] all utilize the transformer structure as the backbone for scene parsing. Among them, SETR achieves much worse parsing results, while other transformer structures obtain superior results than the convolutional backbone. Among them, UperNet [80] using Swin [56] backbone performs much better than the ResNet-50 [32] backbone in terms of both the mIoU and iIoU metrics. Moreover, compared with Swin-UperNet, SegNeXt obtains a similar performance with only about 20% parameters and 7% GFlops.

In summary, the encoder-decoder structure, spatial self-attention mechanism, and transformer structure are very useful strategies for improving the traffic monitoring scene parsing. In Sec. 7, according to the strategies, we design a more powerful decoder that can further improve SegNeXt, which performs better than the Hamburger decoder [26].

Table 4. Evaluation results of previous instance segmentation approaches on TSP6K validation and test sets. We run all the methods based on the ResNet-50 [32] backbone.

Methods	Validation		Test	
	AP_{box}	AP_{seg}	AP_{box}	AP_{seg}
YOLACT [4]	19.9	13.7	20.7	14.6
Mask-RCNN [31]	27.2	23.5	27.0	23.5
SOLO [75]	–	29.6	–	29.8
SOLOv2 [76]	–	28.6	–	28.6
QueryInst [23]	37.7	31.5	37.2	31.3
Mask2Former [14]	32.9	31.3	32.5	31.4

5. Evaluating Instance Segmentation Methods

We provide each traffic image in TSP6K with additional instance annotations, which can be used to evaluate the performance of the instance segmentation methods on segmenting and classifying traffic participants (*i.e.* humans and vehicles) in the traffic monitoring images. The categories of the traffic participants are as follows: person, rider, car, truck, bus, motorcycle, and bicycle. We evaluate several classic instance segmentation methods including YOLACT [4], Mask RCNN [31], SOLO [75], SOLOv2 [76], QueryInst [23] and

Mask2Former [14]. All the above methods are conducted based on the publicly available codebase, mmdetection [8]. The average precision (AP) metric is reported in Tab. 4.

Performance Analysis: Among the evaluated methods, QueryInst [23] achieves superior performance than other methods. It also achieves the best 7.4% AP_s score. The poor performance indicates the existing methods struggle in small instance segmentation. Furthermore, we observe that Mask-RCNN based on ResNet-50 achieves 40.9% box AP and 36.4% mask AP on Cityscapes, which exceed the models trained on TSP6K by more than 10% AP. The performance discrepancy demonstrates instance segmentation on TSP6K is still an enormous challenge. We hope the additional instance annotations aid the community to improve the performance of the instance segmentation methods for segmenting the traffic participants in the monitoring scenes.

Table 5. Evaluation of the unsupervised domain adaption methods.

Methods	SYNTHIA → TSP6K		Cityscapes → TSP6K	
	mIoU (%)	Imprv (%)	mIoU (%)	Imprv (%)
Baseline	21.7	0	26.1	0
ADVENT [73]	22.3	+0.6	31.7	+5.6
DA-SAC [2]	33.0	+11.3	33.9	+7.8
SePiCo [81]	33.8	+12.1	35.9	+9.8
DAFormer [37]	33.4	+11.7	39.5	+13.4
HRDA [38]	45.4	+23.7	54.1	+18.0

6. Unsupervised Domain Adaption

UDA methods for scene parsing are widely studied in recent years. However, most UDA methods focus on adapting the synthetic driving scenes to the real driving scenes. Benefiting from the proposed TSP6K dataset, we can study the UDA methods for adapting the driving scenes to the traffic monitoring scenes. Specifically, we conducted UDA experiments for adapting SYNTHIA [63] and Cityscapes [17] datasets to the TSP6K dataset, respectively. We select several classic UDA methods and evaluate them, including ADVENT [73], DA-SAC [2], SePiCo [81], DAFormer [37], and HRDA [38]. The experiment results are shown in Tab. 5. Note we only count and average the results of the common classes in both the source and target domains.

Performance Analysis: We build a baseline, which trains DeepLab-v2 [10] on the source domain, and directly inferences on the target domain. Compared with the baseline, all evaluated UDA methods outperform it by a large margin. We can observe that the recent transformer-based UDA methods achieve better performance than CNN-based UDA methods. The best performance of UDA methods is still far inferior to the fully-supervised methods (54.1% *vs* 72.4%). Furthermore, UDA from Cityscapes to TSP6K achieves much higher performance than UDA from SYNTHIA to TSP6K. This fact demonstrates the existing driving datasets can facilitate the traffic monitoring scene understanding. We hope that the proposed dataset can facilitate the development of UDA methods for the task of traffic monitoring scene parsing.

7. Proposed Scene Parsing Method

As analyzed in Sec. 3, the traffic monitoring scenes usually capture much more traffic content than the driving scenes and the scale and shape variances of different semantic regions are much larger. Moreover, small things and stuff take a large proportion. These situations make accurately parsing the scenes challenging. To adapt to the traffic monitoring scenes, we propose a detail refining decoder. The design principles of our decoder are two-fold.

First, as the spatial resolution of the last features from the backbone are very low, building decoders based on the low-resolution features usually generates coarse parsing results and, hence largely affects the small object parsing. As verified in some previous works [53, 68, 75], the low-level high-resolution features are helpful for segmenting small objects. Thus, we utilize the encoder-decoder structure to fuse the low-resolution and high-resolution features to improve the small object parsing. **Second**, as analyzed in Sec. 4, self-attention is an efficient way to encode spatial information for scene parsing. However, directly applying the self-attention mechanism to encode high-resolution features will consume massive computation resources, especially when processing high-resolution traffic scene images. Inspired by [14] that learns representations for each segment region, we propose to introduce several region tokens and build pairwise correlations between each region token and each patch tokens from the high-resolution features.

7.1. Overall Pipeline

We construct the scene parsing network for traffic monitoring scenes based on the valuable tips summarized in Sec. 4. First, we adopt the powerful encoder presented in SegNeXt [27] as our encoder, which achieves good results with low computational costs on our TSP6K dataset. Then, we build a detail refining decoder (DRD) upon the encoder. The pipeline of the detail refining decoder is shown in Fig. 4, which contains two parts. For the first part, we follow the decoder design of DeepLabv3+ [12] to generate fine-level feature maps. The ASPP module is added to the encoder directly. Note that we do not use the $\times 4$ downsampling features from the second stage but the $\times 8$ ones from the third stage as suggested in [27]. The second part is the region refining module, which is described in the following subsection.

7.2. Region Refining Module

The region refining module is proposed to refine different semantic regions in the traffic image. Formally, let $\mathbf{F} \in \mathbb{R}^{HW \times C}$ denote the flattened features from the first part

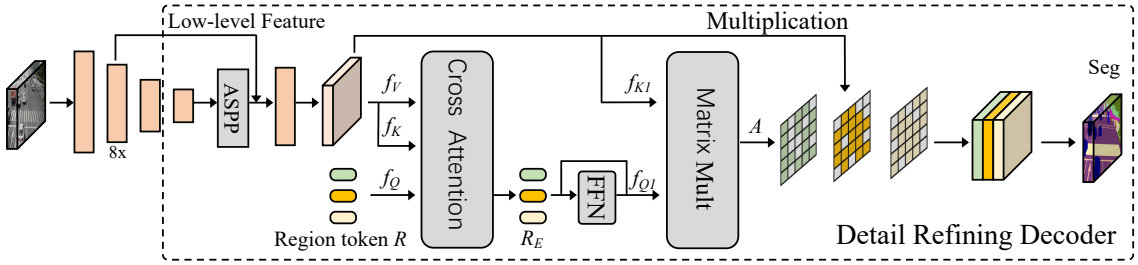


Figure 4. Pipeline of the detail refining decoder. Our decoder contains two parts. The first part is similar to the decoder presented in DeeplabV3+ [12]. Differently, we use the feature maps from the third stage ($\times 8$ downsampling compared to the input) to fuse the feature maps from ASPP. The second part is the proposed region refining module.

of the decoder, where H, W, and C denote the height, width, and the number of channels, respectively. Let $\mathbf{R} \in \mathbb{R}^{N \times C}$ denote N learnable region tokens, each of which is a C-dimensional vector. The flattened features $\mathbf{F}$ and the learnable region tokens $\mathbf{R}$ are separately sent into three linear layers to generate the query, key, and value as follows:

$$\mathbf{R}_Q, \mathbf{F}_K, \mathbf{F}_V = f_Q(\mathbf{R}), f_K(\mathbf{F}), f_V(\mathbf{F}), \tag{1}$$

where $f_Q(\mathbf{R})$, $f_K(\mathbf{F})$, and $f_V(\mathbf{F})$ are linear layers and $\mathbf{R}_Q \in \mathbb{R}^{N \times C}$, $\mathbf{F}_K \in \mathbb{R}^{HW \times C}$, $\mathbf{F}_V \in \mathbb{R}^{HW \times C}$. We compute the multi-head cross-attention between $\mathbf{F}$ and $\mathbf{R}$ as follows:

$$\mathbf{R}_E = \text{Softmax}\left(\frac{\mathbf{R}_Q \mathbf{F}_K^T}{\sqrt{C}}\right)\mathbf{F}_V + \mathbf{R}, \tag{2}$$

where $\mathbf{R}_E \in \mathbb{R}^{N \times C}$ is the resulting region embeddings. The region embeddings are then sent into a feed-forward network, which is formulated as:

$$\mathbf{R}_O = \text{FFN}(\mathbf{R}_E) + \mathbf{R}_E, \tag{3}$$

where $\mathbf{R}_O$ is the output of the feed-forward network. Here, following [69], only the region tokens $\mathbf{R}_E$ are sent to the feed-forward block for an efficient process.

Next, $\mathbf{R}_O$ and $\mathbf{F}$ are delivered to two linear layers to generate a group of new queries and keys as follows:

$$\mathbf{R}_{Q1}, \mathbf{F}_{K1} = f_{Q1}(\mathbf{R}_O), f_{K1}(\mathbf{F}). \tag{4}$$

We perform the matrix multiplication between $\mathbf{R}_{Q1}$ and $\mathbf{F}_{K1}$ to produce attention maps by

$$\mathbf{A} = \text{Softmax}\left(\frac{\mathbf{R}_{Q1} \mathbf{F}_{K1}^T}{\sqrt{C}}\right), \tag{5}$$

where $\mathbf{A} \in \mathbb{R}^{N \times HW}$ denotes N attention maps and each attention map is associated with a semantic region. When we attain the region attention maps, we combine $\mathbf{A}$ and $\mathbf{F} \in \mathbb{R}^{HW \times C}$ via broadcast multiplications, which can be written as follows:

$$\mathbf{S}_{i,j,k} = \mathbf{A}_{i,j} \cdot \mathbf{F}_{j,k}, \tag{6}$$

where $\mathbf{S} \in \mathbb{R}^{N \times HW \times C}$ is the output. Finally, $\mathbf{S}$ is permuted and reshaped to $\mathbb{R}^{N \times C \times H \times W}$, and then sent into a convolutional layer to generate the final segmentation maps.

8. Experiments

Tab. 3 lists the performance of different methods. It can be seen that our method outperforms all previous methods and achieves the best results in terms of both two metrics. To verify the effectiveness of the proposed detail refining decoder, we conduct several ablation experiments on the number of region tokens, and attention heads. Due to the space limit, we put the experimental details and ablation results in the supplementary materials.

9. Conclusions

In this paper, we have constructed the TSP6K dataset, focusing on the traffic monitoring scenes. We have provided each traffic image with a semantic and instance label. Based on the finely annotated TSP6K dataset, we have also evaluated a few popular scene parsing methods, instance segmentation methods, and UDA methods. To improve the performance of the scene parsing, we design a detail refining decoder, which utilizes the high-resolution features from the encoder-decoder structure and refines different semantic regions based on the region refining module. The detail refining decoder learns several region tokens and computes attention maps for different semantic regions. The attention maps are used to refine the pixel affinity in different semantic regions. Experiments have shown the effectiveness of the detail refining decoder.

Limitations: The dataset contains 6,000 labeled images. We still have a large amount of images remaining unlabeled, which can be further explored. The dataset is not diverse geographically, which lacks scenes from the left-hand driving countries. The modality of TSP6K only contains RGB images, which limits the development of multi-modal models.

Acknowledgments: This work was in part supported by National Key R&D Program of China (No. 2022ZD0118700), NSFC (NO. 62276145), the Fundamental Research Funds for the Central Universities (Nankai University, 070-63223049), CAST through Young Elite Scientist Sponsorship Program (No. YESS20210377). Computations were supported by the Supercomputing Center of Nankai University (NKSC).

References

[1] Aau rainsnow traffic surveillance dataset, 2018. 4

[2] Nikita Araslanov and Stefan Roth. Self-supervised augmentation consistency for adapting semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 15384–15394, 2021. 7

[3] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(12):2481–2495, 2017. 3

[4] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *Int. Conf. Comput. Vis.*, pages 9157–9166, 2019. 3, 6

[5] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1209–1218, 2018. 1

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Eur. Conf. Comput. Vis.*, pages 213–229. Springer, 2020. 6

[7] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4974–4983, 2019. 3

[8] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 7

[9] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *Int. Conf. Learn. Represent.*, 2015. 3

[10] 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. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(4):834–848, 2017. 3, 7

[11] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 5, 6

[12] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Eur. Conf. Comput. Vis.*, pages 801–818, 2018. 3, 5, 6, 7, 8

[13] Bowen Cheng, Liang-Chieh Chen, Yunchao Wei, Yukun Zhu, Zilong Huang, Jinjun Xiong, Thomas S Huang, Wen-Mei Hwu, and Honghui Shi. Spgnet: Semantic prediction guidance for scene parsing. In *Int. Conf. Comput. Vis.*, pages 5218–5228, 2019. 1, 3

[14] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1290–1299, 2022. 3, 6, 7

[15] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Adv. Neural Inform. Process. Syst.*, 34:17864–17875, 2021. 3

[16] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 5

[17] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3213–3223, 2016. 1, 2, 3, 4, 5, 7

[18] Jonathan Crespo, Jose Carlos Castillo, Oscar Martinez Mozos, and Ramon Barber. Semantic information for robot navigation: A survey. *Applied Sciences*, 10(2):497, 2020. 1

[19] Dengxin Dai and Luc Van Gool. Dark model adaptation: Semantic image segmentation from daytime to nighttime. In *Proc. Int. Conf. Intelligent Transportation Systems*, pages 3819–3824, 2018. 2

[20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Int. Conf. Learn. Represent.*, 2021. 3, 6

[21] Boyang Du, Congju Du, and Li Yu. Megf-net: multi-exposure generation and fusion network for vehicle detection under dim light conditions. *Visual Intelligence 1*, 2023. 1

[22] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vis.*, 111(1):98–136, 2015. 2

[23] Yuxin Fang, Shusheng Yang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Instances as queries. In *Int. Conf. Comput. Vis.*, pages 6910–6919, 2021. 3, 6, 7

[24] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3146–3154, 2019. 5, 6

[25] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 1, 2, 4

[26] Zhengyang Geng, Meng-Hao Guo, Hongxu Chen, Xia Li, Ke Wei, and Zhouchen Lin. Is attention better than matrix decomposition? In *Int. Conf. Learn. Represent.*, 2021. 6

[27] Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, Zhengning Liu, Ming-Ming Cheng, and Shi-Min Hu. Segnext: Rethinking convolutional attention design for semantic segmentation. In *Adv. Neural Inform. Process. Syst.*, 2022. 2, 3, 6, 7

[28] Meng-Hao Guo, Cheng-Ze Lu, Zheng-Ning Liu, Ming-Ming Cheng, and Shi-Min Hu. Visual attention network. *Computational visual media*, 2023. 3

[29] Meng-Hao Guo, Tian-Xing Xu, Jiang-Jiang Liu, Zheng-Ning Liu, Peng-Tao Jiang, Tai-Jiang Mu, Song-Hai Zhang, Ralph R Martin, Ming-Ming Cheng, and Shi-Min Hu. Attention mechanisms in computer vision: A survey. *Computational visual media*, 8(3):331–368, 2022. 3

[30] Junjun He, Zhongying Deng, Lei Zhou, Yali Wang, and Yu Qiao. Adaptive pyramid context network for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7519–7528, 2019. 1

[31] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Int. Conf. Comput. Vis.*, pages 2961–2969, 2017. 3, 6

[32] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 770–778, 2016. 6

[33] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *Int. Conf. Mach. Learn.*, pages 1989–1998. Pmlr, 2018. 3

[34] Weixiang Hong, Zhenzhen Wang, Ming Yang, and Junsong Yuan. Conditional generative adversarial network for structured domain adaptation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1335–1344, 2018. 3

[35] Qibin Hou, Cheng-Ze Lu, Ming-Ming Cheng, and Jiashi Feng. Conv2former: A simple transformer-style convnet for visual recognition. *arXiv preprint arXiv:2211.11943*, 2022. 3

[36] Qibin Hou, Li Zhang, Ming-Ming Cheng, and Jiashi Feng. Strip pooling: Rethinking spatial pooling for scene parsing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4003–4012, 2020. 3

[37] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9924–9935, 2022. 7

[38] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Hrda: Context-aware high-resolution domain-adaptive semantic segmentation. In *Eur. Conf. Comput. Vis.*, pages 372–391. Springer, 2022. 5, 7

[39] Xinyu Huang, Peng Wang, Xinjing Cheng, Dingfu Zhou, Qichuan Geng, and Ruigang Yang. The apolloscape open dataset for autonomous driving and its application. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(10):2702–2719, 2019. 2

[40] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6409–6418, 2019. 3

[41] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Int. Conf. Comput. Vis.*, pages 603–612, 2019. 3, 5, 6

[42] Galadrielle Humblot-Renaux, Letizia Marchegiani, Thomas B Moeslund, and Rikke Gade. Navigation-oriented scene understanding for robotic autonomy: Learning to segment driveability in egocentric images. *IEEE Robotics and Automation Letters*, 7(2):2913–2920, 2022. 1

[43] Rui Jiang, Ruixiang Zhu, Hu Su, Yinlin Li, Yuan Xie, and Wei Zou. Deep learning-based moving object segmentation: Recent progress and research prospects. *Machine Intelligence Research*, 20(3):335–369, 2023. 1

[44] Xiaojie Jin, Huaxin Xiao, Xiaohui Shen, Jimei Yang, Zhe Lin, Yunpeng Chen, Zequn Jie, Jiashi Feng, and Shuicheng Yan. Predicting scene parsing and motion dynamics in the future. In *Adv. Neural Inform. Process. Syst.*, volume 30, 2017. 1

[45] Jean-Philippe Jodoin, Guillaume-Alexandre Bilodeau, and Nicolas Saunier. Urban tracker: Multiple object tracking in urban mixed traffic. In *IEEE Winter Conf. Appl. Comput. Vis.*, pages 885–892. IEEE, 2014. 4

[46] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 2

[47] Ruihuang Li, Shuai Li, Chenhang He, Yabin Zhang, Xu Jia, and Lei Zhang. Class-balanced pixel-level self-labeling for domain adaptive semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11593–11603, 2022. 3

[48] Xia Li, Zhisheng Zhong, Jianlong Wu, Yibo Yang, Zhouchen Lin, and Hong Liu. Expectation-maximization attention networks for semantic segmentation. In *Int. Conf. Comput. Vis.*, pages 9167–9176, 2019. 5, 6

[49] Zhetao Li, Ziwen Chen, Wei-Shi Zheng, Sangyoon Oh, and Kien Nguyen. Ar-cnn: an attention ranking network for learning urban perception. *Science China Information Sciences*, 65(1):112104, 2022. 1

[50] Qing Lian, Fengmao Lv, Lixin Duan, and Boqing Gong. Constructing self-motivated pyramid curriculums for cross-domain semantic segmentation: A non-adversarial approach. In *Int. Conf. Comput. Vis.*, pages 6758–6767, 2019. 3

[51] Xiaodan Liang, Hongfei Zhou, and Eric Xing. Dynamic-structured semantic propagation network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 752–761, 2018. 1

[52] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1925–1934, 2017. 1, 3

[53] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2117–2125, 2017. 7

[54] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*, 2014. 2

[55] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8759–8768, 2018. 3

[56] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Int. Conf. Comput. Vis.*, pages 10012–10022, 2021. 6

[57] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3431–3440, 2015. 1, 3, 5, 6

[58] Yisheng Lv, Yanjie Duan, Wenwen Kang, Zhengxi Li, and Fei-Yue Wang. Traffic flow prediction with big data: a deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 16(2):865–873, 2014. 1

[59] Luke Melas-Kyriazi and Arjun K Manrai. Pixmatch: Unsupervised domain adaptation via pixelwise consistency train-

ing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12435–12445, 2021. 3

[60] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Int. Conf. Comput. Vis.*, pages 4990–4999, 2017. 2, 4

[61] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *Eur. Conf. Comput. Vis.*, pages 102–118. Springer, 2016. 3

[62] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 1

[63] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3234–3243, 2016. 3, 7

[64] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In *Int. Conf. Comput. Vis.*, pages 7374–7383, 2019. 2

[65] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *Int. Conf. Comput. Vis.*, pages 10765–10775, 2021. 2, 4

[66] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Int. Conf. Comput. Vis.*, pages 7262–7272, 2021. 1, 3

[67] Zheng Tang, Milind Naphade, Ming-Yu Liu, Xiaodong Yang, Stan Birchfield, Shuo Wang, Ratnesh Kumar, David Anastasiu, and Jenq-Neng Hwang. Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8797–8806, 2019. 4

[68] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *Eur. Conf. Comput. Vis.*, pages 282–298. Springer, 2020. 3, 7

[69] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Int. Conf. Comput. Vis.*, pages 32–42, 2021. 8

[70] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7472–7481, 2018. 3

[71] Yi-Hsuan Tsai, Kihyuk Sohn, Samuel Schulter, and Manmohan Chandraker. Domain adaptation for structured output via discriminative patch representations. In *Int. Conf. Comput. Vis.*, pages 1456–1465, 2019. 3

[72] Girish Varma, Anbumani Subramanian, Anoop Namboodiri, Manmohan Chandraker, and CV Jawahar. Idd: A dataset for exploring problems of autonomous navigation in unconstrained environments. In *IEEE Winter Conf. Appl. Comput. Vis.*, pages 1743–1751. IEEE, 2019. 2

[73] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2517–2526, 2019. 3, 7

[74] Haoran Wang, Tong Shen, Wei Zhang, Ling-Yu Duan, and Tao Mei. Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation. In *Eur. Conf. Comput. Vis.*, pages 642–659. Springer, 2020. 3

[75] Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. Solo: Segmenting objects by locations. In *Eur. Conf. Comput. Vis.*, pages 649–665. Springer, 2020. 3, 6, 7

[76] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic and fast instance segmentation. *Adv. Neural Inform. Process. Syst.*, 33:17721–17732, 2020. 3, 6

[77] Yuxi Wang, Junran Peng, and ZhaoXiang Zhang. Uncertainty-aware pseudo label refinery for domain adaptive semantic segmentation. In *Int. Conf. Comput. Vis.*, pages 9092–9101, 2021. 3

[78] Dong Wu, Man-Wen Liao, Wei-Tian Zhang, Xing-Gang Wang, Xiang Bai, Wen-Qing Cheng, and Wen-Yu Liu. Yolop: You only look once for panoptic driving perception. *Machine Intelligence Research*, 19(6):550–562, 2022. 1

[79] Huikai Wu, Junge Zhang, Kaiqi Huang, Kongming Liang, and Yu Yizhou. Fastfcn: Rethinking dilated convolution in the backbone for semantic segmentation. In *arXiv preprint arXiv:1903.11816*, 2019. 3

[80] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Eur. Conf. Comput. Vis.*, pages 418–434, 2018. 5, 6

[81] Binhui Xie, Shuang Li, Mingjia Li, Chi Harold Liu, Gao Huang, and Guoren Wang. Sepico: Semantic-guided pixel contrast for domain adaptive semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2023. 7

[82] Enze Xie, Peize Sun, Xiaoge Song, Wenhai Wang, Xuebo Liu, Ding Liang, Chunhua Shen, and Ping Luo. Polarmask: Single shot instance segmentation with polar representation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12193–12202, 2020. 3

[83] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inform. Process. Syst.*, 34:12077–12090, 2021. 1, 3, 6

[84] Yanchao Yang and Stefano Soatto. Fda: Fourier domain adaptation for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4085–4095, 2020. 3

[85] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. *Int. J. Comput. Vis.*, pages 1–18, 2021. 3

[86] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Eur. Conf. Comput. Vis.*, pages 325–341, 2018. 3

[87] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *IEEE Conf. Comput. Vis. Pattern*

Recog., pages 2636–2645, 2020. 1, 2, 4

[88] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In *Eur. Conf. Comput. Vis.*, 2020. 6

[89] Yuhui Yuan, Lang Huang, Jianyuan Guo, Chao Zhang, Xilin Chen, and Jingdong Wang. Ocnet: Object context for semantic segmentation. *Int. J. Comput. Vis.*, 129(8):2375–2398, 2021. 3

[90] Oliver Zendel, Katrin Honauer, Markus Murschitz, Daniel Steininger, and Gustavo Fernandez Dominguez. Wilddash-creating hazard-aware benchmarks. In *Eur. Conf. Comput. Vis.*, pages 402–416, 2018. 2, 4

[91] Oliver Zendel, Matthias Schörghuber, Bernhard Rainer, Markus Murschitz, and Csaba Beleznai. Unifying panoptic segmentation for autonomous driving. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 21351–21360, 2022. 2

[92] Hang Zhang, Kristin Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Ambrish Tyagi, and Amit Agrawal. Context encoding for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7151–7160, 2018. 5, 6

[93] Qiming Zhang, Jing Zhang, Wei Liu, and Dacheng Tao. Category anchor-guided unsupervised domain adaptation for semantic segmentation. *Adv. Neural Inform. Process. Syst.*, 32, 2019. 3

[94] Rui Zhang, Sheng Tang, Yongdong Zhang, Jintao Li, and Shuicheng Yan. Scale-adaptive convolutions for scene parsing. In *Int. Conf. Comput. Vis.*, pages 2031–2039, 2017. 1

[95] Wenqiang Zhang, Zilong Huang, Guozhong Luo, Tao Chen, Xinggang Wang, Wenyu Liu, Gang Yu, and Chunhua Shen. Topformer: Token pyramid transformer for mobile semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12083–12093, 2022. 3

[96] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-net: Towards unified image segmentation. In *Adv. Neural Inform. Process. Syst.*, volume 34, pages 10326–10338, 2021. 6

[97] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia. Icnet for real-time semantic segmentation on high-resolution images. In *Eur. Conf. Comput. Vis.*, pages 405–420, 2018. 3

[98] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2881–2890, 2017. 1, 3, 6

[99] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. Psanet: Point-wise spatial attention network for scene parsing. In *Eur. Conf. Comput. Vis.*, pages 267–283, 2018. 3, 6

[100] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6881–6890, 2021. 1, 3, 5, 6

[101] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 633–641, 2017. 1, 2

[102] Zhen Zhu, Mengde Xu, Song Bai, Tengteng Huang, and Xiang Bai. Asymmetric non-local neural networks for semantic segmentation. In *Int. Conf. Comput. Vis.*, pages 593–602, 2019. 1

[103] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Eur. Conf. Comput. Vis.*, pages 289–305, 2018. 3

[104] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. Confidence regularized self-training. In *Int. Conf. Comput. Vis.*, pages 5982–5991, 2019. 3