DDC-Net: Semantic Segmentation For Urban Roads Based On Improved Capsule Networks

Xianlei Ge[1,2], Xiaobo Shen[1,3*], and Yingxuan Zhou[1]

[1] School of Electronic Engineering, Huainan Normal University, Huainan 232038, China
[2] College of Computing and Information Technologies, National University, Manila 1008, Philippines
[3] College of Industrial Education, Technological University of the Philippines, Manila 1000, Philippines
[*] Corresponding author. E-mail: shenxb@hnnu.edu.cn

Received: Jan. 06, 2024; Accepted: Dec. 18, 2024

In recent years, with the gradual progress of automatic driving technology, semantic segmentation of road scenes, as the core of this technology, has become a hot spot of research. However, nowadays, most of the convolutional (CNN)-based methods appear to be inefficient and costly due to the factors of large amount of detection data and complex structure. It limits their performance in dealing with some fast response (real-time) tasks. Addressing the above problems, this paper proposes a capsule network-based semantic segmentation method for road images, which achieves a good balance between recognition efficiency and detection speed. Specifically, the DDC-Net designed based on capsule network is used as the baseline network, and different connection paths are dynamically selected according to pixel affinity during forward propagation. In addition, DDC-S and DDC-G are designed for spatial detail fusion and semantic fusion, respectively, and the local feature extraction module (LFCE) is designed using a two-branch structure. Numerous experiments show that the method described in this paper outperforms most of the current CNN-based methods in terms of model size, recognition flexibility and overall performance. In ADE20K and Cityscapes test datasets, the method described in this paper achieves 74.5% and 79.4% mean intersection and merger ratio (mIoU) accuracies at 63.9fps and 64.8fps, and the experimental results demonstrate the effectiveness of our method.

Keywords: image semantic segmentation; deep learning; autonomous driving; road scene detection; fast response

http://dx.doi.org/10.6180/jase.202510_28(10).0009

1. Introduction

Road image semantic segmentation [1] is an important part of automatic driving, aiming at distinguishing and labeling the category to which each pixel location in a road image belongs. In the past, road detection mainly relied on researchers to manually label the data, and the performance is more dependent on the accuracy of the labeling, which is less efficient and susceptible to the subjective influence of the staff. Nowadays, scene detection based on computer vision techniques has achieved remarkable results and has been widely used For example, in the fields of image understanding [2], automated driving [3], medical image analysis [4], and intelligent assistance [5].

With the prominence of deep learning research, Convolutional Neural Networks (CNNs) are gradually becoming the prime choice for dealing with scene detection. The introduction of fully convolutional network (FCN) [6] highlights the limitations of manual-based methods (For example, lack of flexibility, high data accuracy requirements). For example, Fu et al. [7] used self-attention to construct a two-branch network, and Zhao et al. [8] proposed PSPNet through pyramid pooling. These methods rely on the ability of CNNs to extract complex features to create more outstanding performance.

However, limitations of CNNs are gradually emerging.

In general, recognition accuracy and detection speed are often considered to be mutually exclusive in the problem. Most of the models tend to use complex network structures in order to achieve high accuracy, with large model sizes, and performance needs to be improved when dealing with some real-time response tasks.

In order to solve the above dilemma, many lightweight frameworks have been proposed, such as Chen et al. [9], Huang et al. [10], Zhao et al. [11], Yu et al. [12], and Paszke et al. [13]. Although the above methods have made some contributions to the model size, the effect has certain limitations.

In this paper, a segmentation model based on capsule network is proposed to achieve a good balance between performance and detection speed. The capsule network is utilized with some of the properties of MobileNetV3 -small to design DDC-Net, the image spatial detail fusion network (DDC-S) and the semantic feature extraction network (DDC-G) are designed from residual fusion, and the above network is designed as a local compression feature extraction module (LFCE) using a deep two-branch structure. Overall, here are our primary contributions:

1. A new semantic segmentation approach for road images is proposed based on capsule networks, which enables the model to obtain denser image features with lower computational cost.

2. To improve the spatial fusion ability of our model, a spatial detail fusion network (DDC-S) is designed using residual shallow convolution, which effectively enhances the spatial feature focusing ability of the model.

3. A semantic feature map fusion network (DDC-G) is proposed to enhance the ability of combining deep semantic information and shallow spatial information of images by utilizing multi-scale image feature mapping.

4. To reduce the model size and enhance the performance of the model processing on real-time tasks, a (Local feature compression extractor, LFCE) module is designed by utilizing a two-branch structure, which compresses the DDC-S and DDC-G structures, and effectively improves the feature extraction efficiency.

It is experimentally demonstrated that the method described in this paper achieves 74.5% and 79.4% mean intersection and merger ratio (mIoU) accuracies at 63.9 fps and 64.8 fps.

This paper's structure continues below. Section II discusses related work. Section III explains in detail the specific implementation of our proposed method, including the specific structure of each section. Section IV gives experimental results and detailed analysis and a series of discussions. Finally, we conclude the paper in Section VI.

2. Relate work

2.1. Semantic segmentation of road images

ith the gradual emergence of computer vision technology. the CNN, with its local awareness [14] and weight sharing [15], has been outstanding in the road semantic segmentation task, gradually replacing manual detection-based as the mainstream. In the past few years, researchers have extracted features by deepening the deep structure of the model network; for example, Chen et al. [16] adapted semantic segmentation of urban scenes by using a large amount of synthetic data. Sun and Li [17] utilized stacked convolutional layers and three branches of ResNet to enhance model performance. Pan et al. [18] connected two deep information networks in tandem, which was used to expand the effective sensory field. While these methods have made some gains in model accuracy, they tend to have more complex network structures and have limited performance when dealing with real-time tasks.

In recent years, several pioneering approaches have been proposed by some researchers to enhance the size of models. Sun et al. [19] proposed a lightweight recognition framework by fusing real-time RGB-D fusion studies. Orsic et al. [20] enlarged the sensory field by fusing shared features with multiple resolutions. Zhang et al. [21] made some gains in the model lightweight enhancement by constructing a modular architecture and using only a few variables for parameterization, and made some gains in the enhancement of model lightweighting. The above methods, although they have some enhancement, still have room for improvement.

In addition, Cheng et al. [22] achieved a more prominent detection speed through a bottom-up approach using a dual ASPP and dual decoder architecture. Borse et al. [23], Chen et al. [24], Xie et al. [25], and Bashkirova et al. [26] utilized the advanced Transformer for the semantic segmentation model framing and achieved better performance. The breakthroughs in performance of these approaches have achieved exciting results. However, the adoption of Transformer to improve the performance will have the result that the model cannot well take into account the lightweight.

Therefore, in this paper, we design DDC-Net based on capsule network by improving the lightweight network MobileNetV3-small to solve the above difficulties.

2.2. Capsule Network

The rise of CNNs in the field of machine vision has been accompanied by some existential flaws. Some of the characteristics of CNNs, such as when the target undergoes physical changes such as translations, rotations, scaling, etc... CNNs do not change the recognition results because of their changes. Secondly, because high-precision semantic features require a deeper network structure, which causes CNNs to lose more spatial details during processing.

o address the above problems, Sabour et al. [27] proposed the idea of capsule networks. The spatial information in the image is encoded as features. According to the length of the activation carriers to represent the predicted probability, using dynamic routing, the dynamic link intercapsule network paths, to realize the multi-layer cascade of image spatial information.

Recently, capsule networks have achieved exciting results in the area of machine vision, Mazzia et al. [28] proposed capsule networks with self-attentive routing by improving dynamic routing and achieved significant results in feature extraction. Rajasegaran et al. [29] used an algorithm for 3D convolution to achieve deep physical attributes on images by reducing the number of Howard et al. [30] achieved good results in medical image segmentation by designing a dual pathway network and realizing the organic combination of CNN and capsule network. The above methods are based on capsule networks, and all of them have made some breakthroughs in model performance.

In this work, we integrate some of the characteristics of capsule networks with DDC-Net to develop a lightweight baseline network with a good balance between accuracy and speed, which improves the model's attention to spatial detail information while ensuring the overall performance of the model.

3. Architecture

Inspired by the distinctive ASPP cascade structure introduced by Kirillov et al. [31], Reiher et al. [32], and Yang et al. [33], and others, this paper proposes a high-performance semantic extraction framework. Illustrated in Fig. 1, the overall model comprises three main components: a lightweight baseline network (DDC-Net), a capsule network (Capsule Networks), and a locally compressed feature information capture module (LFCE).

The DDC-Net as a whole uses a pyramid structure for convolution and cascading between capsule layers. The upper half is downsampled through the cascade and passed into the CAP layer for spatial information capture. Then, it is passed into the LFCE for the fusion of contextual feature information and spatial information to complete the link output. The rest of this chapter describes the three components of the method in detail.

3.1. Capsule Baseline Network

The baseline network of the model (DDC-Net) is a fusion network that replaces part of the convolution in MobileNetV3-small, which has completed pre-training on the ImageNet dataset, with a capsule.

The DDC-Net convolutional part uses MobileNetV3-small as the main framework. The original structure is shown in Table 1. MobileNetV3, as the latest lightweight classification skeleton nowadays, possesses the advantages of small number of parameters and small model size. It makes it more suitable for fast response real-time tasks. After the accumulation of the first two generations of V1 and V2, V3 inherits the residual structure of deeply separable convolution and linear bottleneck in V1 and V2, and introduces the SE channel attention mechanism and NAS (network architecture search) parameter acquisition module. In addition, a more efficient HEAD is designed in MobileNetV3, which contains a convolutional layer, a global average pooling layer and a fully connected layer, further reducing the number of parameters and computation. This enables V3 to perform well in several vision tasks, especially in environments with limited computational resources such as mobile and embedded devices.

However, as a classification target network, MobileNetV3's output of feature maps in the original structure differs from the semantic segmentation accuracy task. Therefore, we appropriately changed the original network structure of MobileNetV3 based on the image feature information (shown in Table 2). The convolutional and pooling layers after the last bottleneck layer in the original MobileNetV3-small structure are discarded to obtain a simplified version with 9 remaining bottleneck layers and 96 output channels.

Obviously, it is intuitive to reduce the depth of the network to increase the receptive field and thus the feature map. However, this would also mean at the same time that the spatial details of the image are also substantially lost. Therefore, we fuse the SE module (Squeeze-and-Excitation module) with Capsule Networks in the bottleneck layer to improve the model's capture of spatial information.

Specifically, the SE module consists of two parts: Squeeze (global information compression) and Excitation (channel recalibration). The SE module in the original structure enhances the expressiveness of the network by adaptively recalibrating the weights of each channel. CAP captures the pose and other relationships of features between

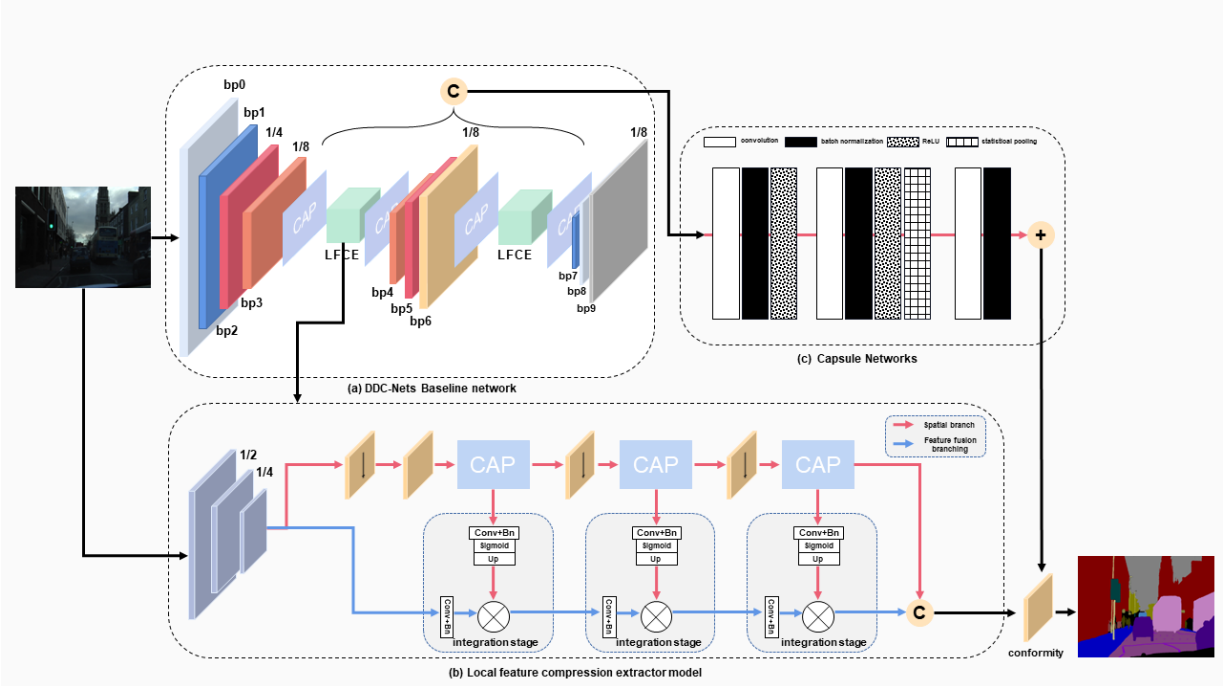

Fig. 1. Overall structure of DDC-Net. Contains three components (a) DDC-Net Baseline network, (b) Locally compressed feature information capture module, and (c) Capsule Networks.

Table 1. Original hierarchy of MobileNetV3-small.

Input	Operator	expsize	#out	SE	NL	S
2242 × 3	conv2d, 3x3	-	16	-	HS	2
1122 × 16	bneck, 3x3	16	16	✓	RE	2
562 × 16	bneck, 3x3	72	24	-	RE	2
282 × 24	bneck, 3 × 3	88	24	-	RE	1
282 × 24	bneck, 5 × 5	96	40	✓	HS	2
142 × 40	bneck, 5 × 5	240	40	✓	HS	1
142 × 40	bneck, 5 × 5	240	40	✓	HS	1
142 × 40	bneck, 5x5	120	48	✓	HS	1
142 × 48	bneck, 5 × 5	144	48	✓	HS	1
142 × 48	bneck, 5 × 5	288	96	✓	HS	2
72 × 96	bneck, 5 × 5	576	96	✓	HS	1
72 × 96	bneck, 5 × 5	576	96	✓	HS	1
72 × 96	conv2d, 1x1	-	576	✓	HS	1
72 × 576	pool, 7 × 7	-	-	-	-	1
12 × 576	conv2d 1x1, NBN	-	1024	-	HS	1
12 × 1024	conv2d 1x1, NBN	-	k	-	-	1

capsules by organizing neurons into capsules. A dynamic routing algorithm passes information between capsules, enhancing the model's ability to capture complex patterns.

For restructuring, in the Squeeze phase, DDC-Net, as in the traditional SE module, uses global average pooling to compress the spatial dimensions of the input feature map into a single channel vector. While in the Excitation phase, the DDC-Net uses the capsule layer of the capsule network instead of the fully connected layer, and CAP generates the attention weights for each channel.

Through the dynamic routing mechanism, the input features are mapped to a high-dimensional space and more complex channel attention is generated. Dynamic routing will determine which channels should be enhanced or suppressed. This will make the SE module more adaptable and able to capture more complex feature relationships.

Table 2. Hierarchical structure of DDC-Net improved version MobileNetV3-small.

Block	Input	Operation	SE	NL	#Out	S
Block0	$224 \times 224 \times 3$	conv2d	-	HS	16	2
Block1	$112 \times 112 \times 16$	bottleneck	✓	RE	16	2
Block2	$56 \times 56 \times 16$	bottleneck	-	RE	24	2
Block3	$28 \times 28 \times 24$	bottleneck	-	RE	24	1
Block4	$28 \times 28 \times 24$	bottleneck	✓	HS	40	2
Block5	$14 \times 14 \times 40$	bottleneck	✓	HS	40	1
Block6	$14 \times 14 \times 40$	bottleneck	✓	HS	48	1
Block7	$14 \times 14 \times 48$	bottleneck	✓	HS	48	1
Block8	$14 \times 14 \times 48$	bottleneck	✓	HS	96	2
Block9	$7 \times 7 \times 96$	bottleneck	✓	HS	96	1

3.1.1. dynamic routing algorithm

In CAP, the type of capsule can be categorized into subtype and parent type, and dynamic routing during forward propagation dynamically links subcapsules based on the output of the parent capsule. In this paper, we use a transformation matrix to transform the vectors of the input capsules to form a ballot and group them with similar votes. These votes eventually become the output vectors of the parent capsule. c_{ij} is the similarity weight (coupling coefficient) between the output vectors, which is calculated as follows:

$$c_{ij} = \frac{\exp\left(b_{ij}\right)}{\sum_K \exp\left(b_{ij}\right)}$$

Where b_{ij} is the similarity of the lower capsule to the previous capsule, which is initialized by default to 0 before each iteration. here we use Softmax to normalize the similarity and finally get the similarity weights. Using Softmax ensures that all weights c_{ij} are nonnegative and sum to one. Essentially, Softmax enforces the probabilistic nature of the coupling coefficients c_{ij}. Conceptually, computing the similarity weights measures how likely a capsule is to activate the capsule.

For all capsules except the first layer capsule, we need the transformation matrix to transform the dimension by multiplying it with the output of the lower layer capsule and obtain the new output (prediction vector) $\hat{u}_{j|i}$ by multiplying the output u_i of the next layer capsule by the weight matrix W_{ji}.

$$\hat{u}_{j|i} = W_{ji} u_i$$

The total input s_j for a capsule is a weighted sum of all similarity weights (coupling coefficients) c_{ji} from the next layer of capsules with all prediction vectors.

$$s_j = \sum c_{ji} \hat{u}_{j|i}$$

We also define the capsule's output vector length to indicate the chance that its entity appears in the current input. Therefore, we ensure that the direction of the capsule's output is preserved by a squash nonlinear function, while the length is restricted to less than one, with the short vector compressed to almost zero and the long vector to little less than one.

$$v_j = \frac{\left\|s_j\right\|^2}{1 + \left\|s_j\right\|^2} \frac{s_j}{\left\|s_j\right\|^2}$$

Intuitively, the prediction vector is a vote from the capsule and influences the output of the capsule. If the activation vector is highly similar to the prediction vector, thereby indicating good correlation between the two capsules. The scalar product of the prediction vector and activation vector measures similarity. b_{ij} is updated and computed as follows:

$$b_{ji} \leftarrow b_{ji} + \hat{u}_{j|i} \cdot v_j$$

As a result, the inter-capsule similarity score takes into account both likelihood and feature attributes instead of neurons focusing only on likelihood. The addition of dynamic routing algorithms allows DDC-Net to utilize fewer parameters and choose the connection paths of the inter-capsule network more flexibly. In addition, the method of dynamically capturing the hierarchical relationships of the features avoids the reduction of the feature map dimensions during the CNN maximum pooling process.

3.2. Local feature compression extractor model

Fig. 2 shows the structure of LFCE, which is composed of two deep neural network branch architectures: a spatial feature information capture branch, and a semantic feature fusion branch. The two branches share the generated feature maps, allowing us to construct segmentation models more quickly. The above two branch networks will be introduced separately in the following.

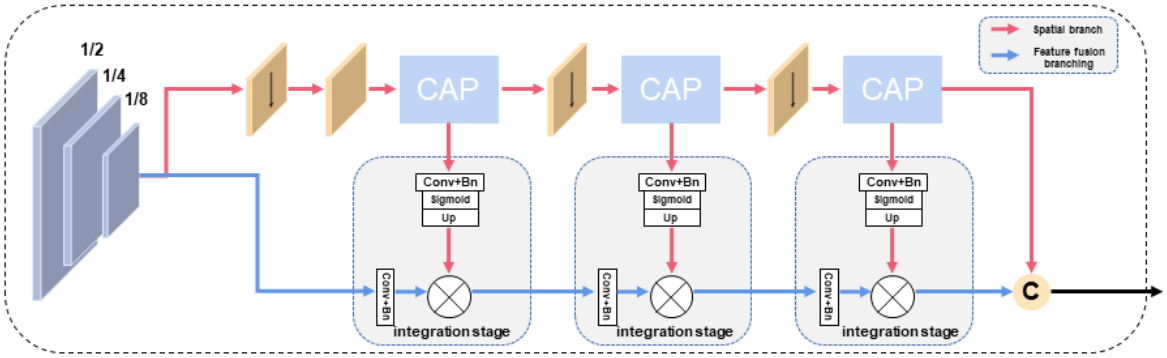

Fig. 2. Overall structure of the Localized Compressed Feature (LFCE) capture module. It contains a spatial branch (red), a semantic feature fusion branch (blue), and symbols representing element multiplication.

3.2.1. *Spatial integration branch*

When dealing with the task of semantic segmentation of road images, some approaches tend to pursue deeper levels of semantic information to fulfill the requirements of high-performance segmentation tasks. This means adopting complex network structures to increase the model acceptance field. However, as the complexity of the structure increases, the ability of the model to capture spatial information is more and more lost.

To address the above problems, we design DDC-S (spatial detail fusion branch). It is connected to the parent capsule in the capsule baseline network, and the spatial information activation probability is calculated based on the activation carriers and affinities for the purpose of spatial information fusion. Specifically, we use the simplified version of ResNet-18 as the baseline network of this branch, and the simplified version of ResNet-18 contains only the first two layers of the original structure of ResNet-18 as the network structure shown in Table 3.

Table 3. LFCE shallow convolutional spatial detail extraction branching hierarchy.

Layer	Input size	Operation	Channel	Stride
layer0	56 × 56	conv2d	64	2
	56 × 56	max pool	-	2
layer1	56 × 56	conv2d	64	1
	56 × 56	conv2d	64	1
	56 × 56	conv2d	64	1
	56 × 56	conv2d	64	1

The output of DDC-S is connected to the parent capsule in DDC-Net as the final output. The number of channels in the final output of this branch is 64. the outputs of the spatial and semantic branches are spliced together and the number of channels is adjusted to 40 by a 1x1 convolutional layer, keeping the feature map size at 28x28 to match the size of the feature map input from Block4.

3.2.2. *Semantic Fusion Branch*

For semantic feature fusion, some approaches [31–33] achieve the effect of combining different input cascade features by linking the attribute convolution and projection convolution of BNs respectively. Inspired by Takikawa et al. [34] for feature fusion, we design the semantic feature fusion branch network (DDC-G).

DDC-G first connects the feature mapping of two branches along the channel when the semantic and spatial shallow information is received. Then, normalization batch processing is applied to shorten the feature distance and balance the feature scale. Specifically, for high to low channel fusion, the high channel feature maps are unsampled by a convolution sequence of size 3×3 and step size 2 before pointwise summation to expand the number of channels to 64 (the output of the feature maps is adjusted to 28×28 using block4 as an example). For low to high, the low channel feature maps are first compressed by 1×1 convolution and then upsampled using bilinear interpolation to reduce the fused structure to the original input image size. The structure is shown in Fig. 3.

4. Experiments

4.1. Datasets

The method proposed in this paper will be evaluated experimentally on ADE20K, CamVid and Cityscapes datasets.

ADE20K (ADE20K Scene Parsing Challenge) is a large-scale dataset widely used in image scene parsing (Semantic Segmentation) research, aiming at advancing semantic seg-

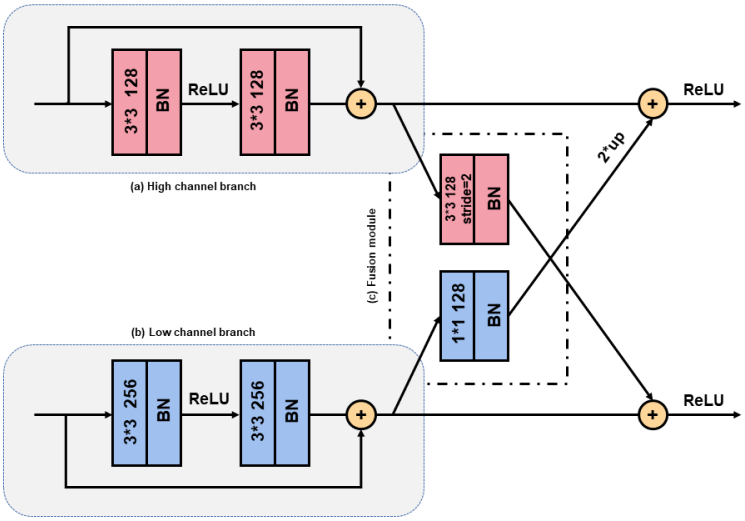

Fig. 3. Semantic feature fusion branch DDC-G bilateral fusion details.

mentation techniques in the field of computer vision in complex scenes. The ADE20K dataset contains about 20,210 images with varying resolutions and sizes, which cover a rich and diverse range of environments, including indoor, outdoor, urban, and rural environments. The ADE20K dataset contains about 20,210 images of different resolutions and sizes, which cover a rich variety of scenes, including indoor, outdoor, urban, and rural environments, etc. The ADE20K dataset is divided into a training set, a validation set and a test set. The training set contains about 20,210 images, the validation set contains about 2,000 images, and the test set contains about 3,000 images. The test set is not provided with labeling and is used for performance evaluation of the model. The images in the dataset are labeled with 150 different object classes and scene classes, including people, vehicles, animals, furniture, natural landscapes, etc. Each pixel is labeled with the category it belongs to for pixel-level semantic segmentation. The multi-category object and scene labeling makes this task a very great challenge in the evaluation of semantic segmentation methods.

CamVid (Cambridge-driving Labeled Video Database) is an image dataset for road scenes, which is widely used in semantic segmentation research in the field of autonomous driving. It consists of about 701 high-quality images of Cambridge city street scenes captured from a car driver's perspective, covering elements such as city roads, vehicles, pedestrians and traffic signs, etc. The CamVid dataset is divided into a training set (367 frames), a validation set (101 frames), and a test set (233 frames), and each frame is accurately labeled on a pixel-level, covering 32 road and city driving-related categories. related categories of road and city driving. Despite its small size, the CamVid dataset plays a key role in image parsing and decision making in automated driving systems, especially in analyzing and understanding complex scenarios in urban road environments.

Cityscapes is a dataset focused on semantic segmentation and object detection tasks in urban scenes. The dataset provides high-resolution images covering a wide range of scenes such as city streets, roads, buildings, etc. The Cityscapes dataset contains about 5,000 high-resolution images with an image size of 1024×2048, which are divided into a training set, a validation set, and a test set. The training set contains 2,975 images, the validation set contains 500 images, and the test set contains 1,525 images. The test set is provided without annotation and is used to evaluate the performance of the model. These images cover street and traffic scenes in real urban environments with different weather, time of day, and seasons, adding to the diversity of the data. The method proposed in this paper will be based on training the training set with the validation set and the trained model will be evaluated on the validation set with common semantic categories of road scenes.

4.2. Training details

The baseline backbone network for the model designed in this paper is based on MobileNetV3-small that has completed pre-training on the ImageNet dataset and fine-tuned it after its training is complete. For a fair comparison we followed the batch size, training iteration scheduling and

data augmentation strategy of Zhao et al. [11]. The initial learning rate was 0.006 , the weights were decayed to 0.0005 , the momentum was set to 0.9 , the batch size was set to 16 , and a "multiple" learning rate with a power of 0.9 was used. In the inference process, the same image resizing and cropping rules as in the above method were set, and the input image size was resized from 1024×2048 to 512×1024 for training, and 1024×512 for testing on the cityscape dataset.

4.3. Experimental environment

Table 4 illustrates the comprehensive configuration details encompassing both hardware and software deployed during the experimental phase.

Table 4. Experiment environment.

Environment	Configuration
CPU	Intel Core i7 12700k
GPU	Nvidia Tesla V100 32GB
Memory	DDR4 16GB
Hard Disc	WestData SSD 1TB
Operating System	Windows11
Python	3.8
torch	1.12 .1

4.4. Analysis of results

4.4.1. Cityscapes

As shown in Table 5, we first evaluate several representative real-time high-performance semantic segmentation methods on the Cityscapes validation dataset, from which we can see that the method in this paper achieves a good balance between detection speed and accuracy, DDC-Net achieves 79.4%mIoU on the validation set with 65 fps , and with similar inference speeds, it is better than SFNet (DF2) and BiSeNet2 by 3.8%mIoU and 6.6%mIoU, respectively. In addition, while maintaining high accuracy, DDC-Net runs 40% faster SwiftNetRN-18 on the Cityscapes validation set and 3 times faster than SFNet (ResNet-18). Facing some accuracyoriented methods (PSPNet, DDRNet-39) although the difference in accuracy is about $1\% - 1.5\%$ mIoU, DDC-Net improves 3-6 times in inference speed in comparison. It is worth noting that although it slightly lags behind some methods (For example, BiSeNet1, DFANet, Fast-SCNN) in terms of inference speed and model size, all of them have a 5%-10% mIoU improvement in terms of model accuracy.

In addition, for the computational complexity (FLOPs), the information that capsule networks in DDC-Net focus on is more oriented to the spatial information of the image, and even though the computational complexity slightly lags behind that of some methods (For example, ICNet, DFANet), it meets the lightweight criterion, and there is a significant increase in the accuracy.

To enhance the credibility of our findings, we have conducted a comprehensive qualitative analysis using the CamVid dataset. This is depicted in Fig. 4, where we present a balanced representation of both successful outcomes and instances of prediction errors. The results underscore the proficiency of DDC-Net in navigating complex urban road scenarios, characterized by varying lighting conditions, shadows, and traffic density. However, it is noteworthy that DDC-Net exhibits certain limitations, particularly in the precision of edge segmentation between closely resembling objects, as exemplified in the third prediction result of Fig. 4. This shortfall is likely attributed to the network's occasional challenges in real-time task execution. A similar trend is observed in other methods prioritizing accuracy, such as PSPNet, indicating a common hurdle in this field of study. It is worth noting that when dealing with intersecting streetscapes, spatial splicing misalignment occurs occasionally due to the structural properties of the capsule network (a failure case is given in Fig. 4). The adjustment for capsule network weights is worth further optimization and adjustment in the future.

4.4.2. ADE20K

To validate the effectiveness of our method in this paper, we also evaluated our method on the ADE20K dataset. This dataset contains about 20,210 images with inconsistent resolution and the validation set contains about 2,000 images. To ensure the fairness of the experimental results, we randomly select 500 images from the validation set for the experiment. For example, the validation results are shown in Table 6, from which we can see that the method in this paper achieves 74.5% mIoU on the validation set at 63.9 fps. which is comparable to SFNet in terms of accuracy. Although DDC-Net is slightly lower than PSPNet in terms of accuracy, it achieves better inference speed. In terms of inference speed, DDC-Net has a 78.4% performance improvement over ICNet, and although there is a speed gap with BiSeNet1, it meets the requirements of real-time tasks and has a 9.4% improvement in accuracy.

4.5. Ablation Experiment

In the previous sections, we have designed a series of methods aimed at improving model efficiency and inference speed, and in this section, we will conduct experiments based on the components (DDC-Net, LFCE) of our proposed methods and present their effectiveness separately.

Table 5. Comparison of the proposed method with other state-of-the-art methods on Cityscapes validation dataset. Our results are in boldface.

Method	Resolution	FLOPs (G)	Params (M)	Time (ms)	Speed (fps)	mIoU (%)
SFNet (DF1) [35]	2048 × 1024	-	9.03	59	90.3	72.1
SFNet (DF2) [35]	2048 × 1024	-	10.53	95	65.2	75.6
SFNet (ResNet-18) [35]	2048 × 1024	2	12.87	123	20.0	78.3
ICNet [11]	2048 × 1024	28.3	26.5	41	31.2	70.1
Fast-SCNN [36]	2048 × 1024	-	1.1	20	133.6	67.5
ERFNet [37]	1024 × 512	27.7	20	26	48.6	69.6
FRRN [38]	1024 × 512	235	-	420	5.2	76.3
CRF-RNN [39]	1024 × 512	-	-	813	2.9	63.1
PSPNet[40]	713 × 713	412.2	250.8	1069	11.0	80.9
BiSeNet1 [12]	1536 × 768	14.8	5.8	14	96.5	69.8
BiSeNet2 [12]	1536 × 768	55.3	49	19	56.2	72.8
SwiftNetRN-18 [41]	2048 × 1024	104.0	24.7	97	46.2	73.1
DFANet [42]	1024 × 1024	3.4	7.8	11	104.3	68.4
DDRNet-39 [43]	2048 × 1024	281.2	32.3	148	23.1	81.3
Ours	**1024 × 512**	**12.8**	**8.67**	**16**	**64.8**	**79.4**

"-" indicates that the method does not provide the corresponding result.

Table 6. Comparison of the proposed method with others on the ADE20K validation dataset. Our results are in bold.

Method	Time (ms)	Speed (fps)	mIoU (%)
SFNet (DF1) [35]	68	-	70.5
SFNet (ResNet-18) [35]	-	18.9	76.4
ICNet [11]	56	35.8	67.9
Fast-SCNN [36]	21	-	64.3
PSPNet [40]	-	13.7	76.8
BiSeNet1 [12]	16	90.6	65.7
SwiftNetRN-18	-	-	68.9
DFANet	17	84.3	62.1
Ours	**21**	**63.9**	**74.5**

"-" indicates that the method does not provide the corresponding result.

4.5.1. Capsule Baseline Network

We selected the network consisting of capsule-based network with MobileNetV3 as the baseline network for the model designed in this paper. To evaluate the effectiveness of this baseline network, two other models are also included in this section, including MobileNetV1, MobileNetV2. In addition, for the fairness of the experiments, MobileNetV3 and MobileNetV2* are added to the experiments for comparison. All the baseline networks are pretrained on ImageNet dataset, after downsampling all the input images to a uniform size (1024 ×512), the feature size is unified using bilinear interpolation with the help of the U-Net structure and evaluated to assess the above-mentioned baseline networks on cityscape validation dataset.

In Table 7, a comparative analysis of MobileNetV1, MobileNetV2, and MobileNetV3, both in their original configurations and when integrated with U-Net, is presented. This analysis reveals a marked enhancement in both accuracy and inference speed with MobileNetV3. Embodying the key attributes of its predecessors, MobileNetV1 and V2,

Table 7. Comparison between speed and accuracy of different baseline network fusion strategies on cityscape validation dataset.

Baseline Network	mIoU (%)	Speed (fps)
U-Net+MobileNetV1	54.3	22.7
U-Net+MobileNetV2	61.7	60.9
U-Net+MobileNetV3	68.4	62.5
MobileNetV2*	69.1	68.9
Ours	71.2	72.6

* Represents partial replacement of convolutional layers in the original structure with capsule network layers.

MobileNetV3 innovatively incorporates multi-scale feature extraction and a channel attention mechanism, leading to a more optimized structural design and model size. Consequently, MobileNetV3 is selected as the foundational network for DDC-Net, given its superior performance metrics and architectural advancements. Fig. 5 provides a detailed comparative analysis of qualitative segmentation performance on the CamVid dataset, employing U-Net combined

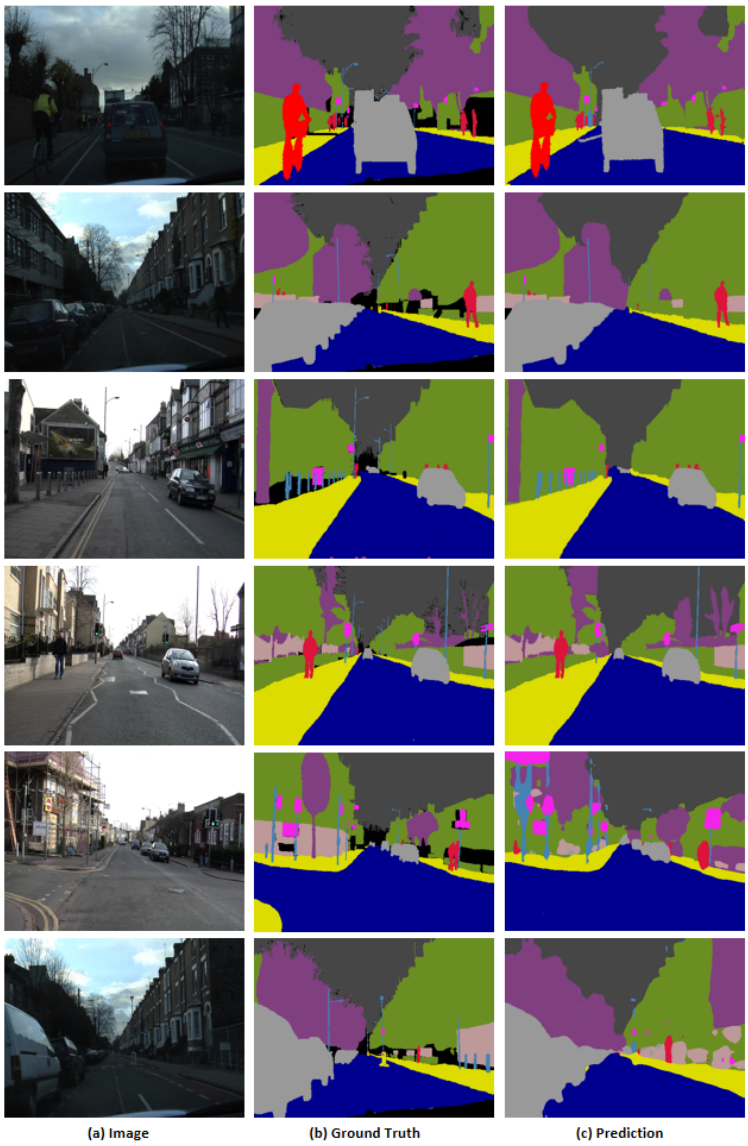

Fig. 4. Visualization results on the CamVid dataset, from left to right, the input image, the real image, and our predicted results. The last two lines show some of the failures.

with MobileNetV2 and an enhanced MobileNetV3 featuring capsule networks.

Compared to MobileNetV3, which uses U-Net with the original structure, DDC-Net improves the accuracy by 2.8% mIoU and the inference speed by about 16% compared to the original structure. Compared to MobileNetV2, which uses U-Net with the original structure, the improved version MobileNetV2 improves 7.4% mIoU in accuracy and 13% in inference speed compared to the original structure.

From the above experiments, the capsule network-based baseline network in this paper not only has an improvement in model accuracy, but also has a significant breakthrough in inference speed.

4.5.2. Local feature compression extractor model

This section focuses on verifying the effectiveness of the Localized Compressive Feature Extraction (LFCE) module. Since LFCE adopts a deep two-branch structure. In order to ensure the fairness of the experiment, this part of the experiment will be combined with the above DDC-Net for the structural and functional experiments. The structural experiment includes the comparison of single-branch structure and the comparison of deep two-branch structure. The functional experiment includes the comparison of the

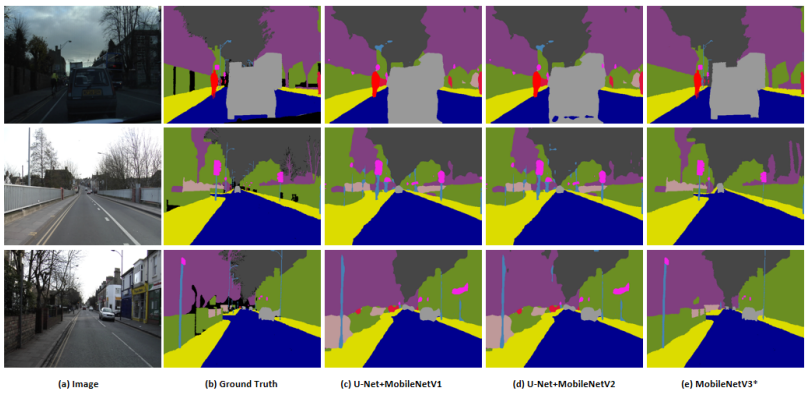

Fig. 5. Results of qualitative segmentation of U-Net+MobileNetV1, U-Net+MobileNetV2 with MobileNetV3* (using capsule networks) on the CamVid dataset.

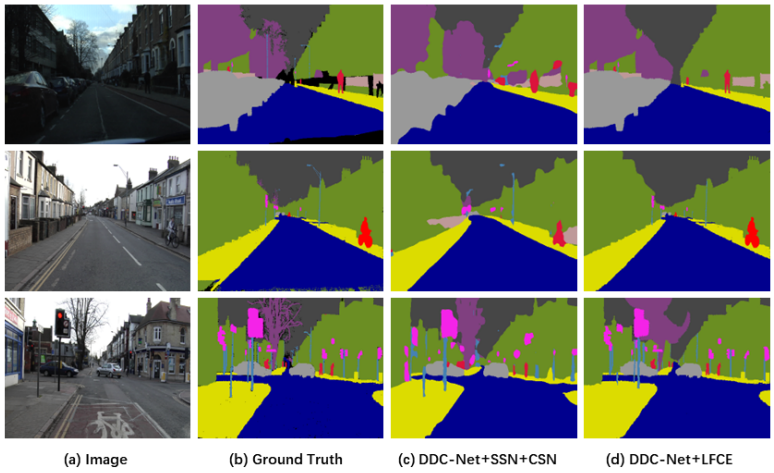

Fig. 6. An Illustrative Comparison of Qualitative Segmentation Performance.

functionality of each branch network. The results of the functional experiments are shown in the following table.

Table 8. Comparison of the rate and accuracy of LFCE's branch-by-branch network and single-branch structural spatial detail extraction network (SSN) with single-branch structural semantic feature fusion structure (CSN) on cityscape validation dataset.

Method	mIoU (%)	Speed (fps)
DDC-Net+SSN	72.8	51.2
DDC-Net+CSN	71.6	67.3
DDC-Net+DDC-S	7 4.6	67.9
DDC-Net+DDC-G	73.9	69.1

From Table 8, we can see that DDC-S improves 1.8% mIoU in accuracy and 32% in inference speed compared to SSN. Compared to the original structure, the inference speed only decreases by about 6% compared to the original baseline network without added branching structure, but the accuracy is improved by 3.4% mIoU. DDC-G improves 2.3% mIoU in accuracy and 2.6% in inference speed compared to CSN. Compared to the original structure, the accuracy is improved by 2.7% mIoU with only 4.8% loss of inference speed. From the above results, it can be seen that DDC-S and DDC-G have achieved improvement in accuracy along with improvement in inference speed compared to SSN and CSN. The functional effectiveness of each branch network of LFCE is verified.

Fig. 6 displays the qualitative segmentation results for multiple structures in the CamVid validation set, highlight-

Table 9. Comparison of the rate and accuracy of the single branch stacked structure with the LFCE deep two-branch structure versus the normal deep convolutional network two-branch structure on the cityscape validation dataset. LUCE denotes the integration of SSN+CSN using the deep two-branch structure.

Method	mIoU (%)	Speed (fps)
DDC-Net+SSN+CSN	74.7	35.4
DDC-Net+LUCE	73.2	40.8
DDC-Net+LFCE	7 9.4	64.8

ing the distinct advantage of DDC-Net+LFCE in detail processing. Table 9 displays structural experiment findings, as can be seen from the data, LUCE improves about 15% in inference speed compared with SSN + CSN. However, due to the local compression of the two-branch structure, which changes part of the structure of SSN and CSN, some of the convolutional layers are not able to output properly, and there is a decrease of 1.5% mIoU in accuracy. However, LFCE with SSN+CSN has an improvement in inference speed of about 85% along with an improvement in accuracy of 4.7%mIoU. Compared with LUCE, the accuracy is improved by 6.2%mIoU and the inference speed is improved by about 58%. Fig. 6 displays the qualitative segmentation results for multiple structures in the CamVid validation set, highlighting the distinct advantage of DDC-Net+LFCE in detail processing. The effectiveness of the shallow convolutional two-branch structure of LFCE is verified.

5. Conclusions

This work introduces a new semantic segmentation method for road images that balances inference speed and efficiency. The designed model consists of two parts: capsule baseline network DDC-Net, and locally compressed feature capture module LFCE. DDC-Net improves the model's ability to maintain spatial details by adjusting the network structure and replacing a part of the convolutional pooling layer with a capsule network layer. In addition, in LFCE, the spatial information fusion branch DDC-S and the semantic fusion branch DDC-G are designed by utilizing the deep two-branch structure for the effective fusion of deep and shallow information. High accuracy is maintained while sustaining the real-time reasoning task. The method presented in this paper has been proven to be effective through our experiments on ADE20K and Cityscapes datasets, both qualitatively and quantitatively.

References

[1] Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew, (2018) *"A review of semantic segmentation using deep neural networks"* **International journal of multimedia information retrieval 7**: 87–93. DOI: 10.1007/s13735-017-0141-z.

[2] Y. Xu, M. Li, L. Cui, S. Huang, F. Wei, and M. Zhou. "Layoutlm: Pre-training of text and layout for document image understanding". In: *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 2020, 1192–1200. DOI: 10.1145/3394486.3403172.

[3] H. Shao, L. Wang, R. Chen, H. Li, and Y. Liu. "Safety-enhanced autonomous driving using interpretable sensor fusion transformer". In: *Conference on Robot Learning*. PMLR. 2023, 726–737. DOI: 10.48550/arXiv.2207.14024.

[4] L. Hou, Y. Cheng, N. Shazeer, N. Parmar, Y. Li, P. Korfiatis, T. M. Drucker, D. J. Blezek, and X. Song. *High Resolution Medical Image Analysis with Spatial Partitioning*. 2019. DOI: 10.48550/arXiv.1909.03108. arXiv: 1909.03108 [eess.IV].

[5] Y. Yao, M. Xu, C. Choi, D. J. Crandall, E. M. Atkins, and B. Dariush. "Egocentric Vision-based Future Vehicle Localization for Intelligent Driving Assistance Systems". In: *2019 International Conference on Robotics and Automation (ICRA)*. 2019, 9711–9717. DOI: 10.1109/ICRA.2019.8794474.

[6] E. Shelhamer, J. Long, and T. Darrell, (2017) *"Fully Convolutional Networks for Semantic Segmentation"* **IEEE Transactions on Pattern Analysis and Machine Intelligence *39***(4): 640–651. DOI: 10.1109/TPAMI.2016.2572683.

[7] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu. "Dual Attention Network for Scene Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019. DOI: 10.1109/CVPR.2019.00326.

[8] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. "Pyramid Scene Parsing Network". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. DOI: 10.1109/CVPR.2017.660.

[9] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, (2018) *"DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs"* **IEEE Transactions on Pattern Analysis and Machine Intelligence *40***(4): 834–848. DOI: 10.1109/TPAMI.2017.2699184.

[10] H. Huang, L. Lin, R. Tong, H. Hu, Q. Zhang, Y. Iwamoto, X. Han, Y.-W. Chen, and J. Wu. "UNet 3+: A Full-Scale Connected UNet for Medical Image Segmentation". In: *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2020, 1055–1059. DOI: 10.1109/ICASSP40776.2020.9053405.

[11] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia. "ICNet for Real-Time Semantic Segmentation on High-Resolution Images". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018. DOI: 10.48550/arXiv.1704.08545.

[12] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang. "BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018. DOI: 10.1007/978-3-030-01261-8_20.

[13] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. *ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation*. 2016. DOI: 10.48550/arXiv.1606.02147. arXiv: 1606.02147 [cs.CV].

[14] X. Ding, C. Xia, X. Zhang, X. Chu, J. Han, and G. Ding. *RepMLP: Re-parameterizing Convolutions into Fully-connected Layers for Image Recognition*. 2022. DOI: 10.48550/arXiv.2105.01883. arXiv: 2105.01883 [cs.CV].

[15] G. Bender, H. Liu, B. Chen, G. Chu, S. Cheng, P.-J. Kindermans, and Q. V. Le. "Can Weight Sharing Outperform Random Architecture Search? An Investigation With TuNAS". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020. DOI: 10.1109/CVPR42600.2020.01433.

[16] Y. Chen, W. Li, and L. Van Gool. "ROAD: Reality Oriented Adaptation for Semantic Segmentation of Urban Scenes". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. DOI: 10.1109/CVPR.2018.00823.

[17] J. Sun and Y. Li, (2021) *"Multi-feature fusion network for road scene semantic segmentation"* **Computers & Electrical Engineering *92***: 107155. DOI: 10.1016/j.compeleceng.2021.107155.

[18] H. Pan, Y. Hong, W. Sun, and Y. Jia, (2023) *"Deep Dual-Resolution Networks for Real-Time and Accurate Semantic Segmentation of Traffic Scenes"* **IEEE Transactions on Intelligent Transportation Systems *24***(3): 3448–3460. DOI: 10.1109/TITS.2022.3228042.

[19] L. Sun, K. Yang, X. Hu, W. Hu, and K. Wang, (2020) *"Real-Time Fusion Network for RGB-D Semantic Segmentation Incorporating Unexpected Obstacle Detection for Road-Driving Images"* **IEEE Robotics and Automation Letters *5***(4): 5558–5565. DOI: 10.1109/LRA.2020.3007457.

[20] M. Orsic, I. Kreso, P. Bevandic, and S. Segvic. "In Defense of Pre-Trained ImageNet Architectures for Real-Time Semantic Segmentation of Road-Driving Images". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019. DOI: 10.1109/CVPR.2019.01289.

[21] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, H. Lin, Z. Zhang, Y. Sun, T. He, J. Mueller, R. Manmatha, M. Li, and A. Smola. "ResNeSt: Split-Attention Networks". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. 2022, 2736–2746. DOI: 10.1109/CVPRW56347.2022.00309.

[22] B. Cheng, M. D. Collins, Y. Zhu, T. Liu, T. S. Huang, H. Adam, and L.-C. Chen. "Panoptic-DeepLab: A Simple, Strong, and Fast Baseline for Bottom-Up Panoptic Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020. DOI: 10.1109/CVPR42600.2020.01249.

[23] S. Borse, Y. Wang, Y. Zhang, and F. Porikli. "InverseForm: A Loss Function for Structured Boundary-Aware Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, 5901–5911. DOI: 10.1109/CVPR46437.2021.00584.

[24] Z. Chen, Y. Duan, W. Wang, J. He, T. Lu, J. Dai, and Y. Qiao. *Vision Transformer Adapter for Dense Predictions*. 2023. DOI: 10.48550/arXiv.2205.08534. arXiv: 2205.08534 [cs.CV].

[25] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo. "SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan. ***34***. Curran Associates, Inc., 2021, 12077–12090. DOI: 10.48550/arXiv.2105.15203.

[26] D. Bashkirova, M. Abdelfattah, Z. Zhu, J. Akl, F. Alladkani, P. Hu, V. Ablavsky, B. Calli, S. A. Bargal, and K. Saenko. "ZeroWaste Dataset: Towards Deformable Object Segmentation in Cluttered Scenes". In: *Proceedings of the IEEE/CVF Conference on Computer Vision*

and Pattern Recognition (CVPR). 2022, 21147–21157. DOI: 10.1109/CVPR52688.2022.02047.

[27] S. Sabour, N. Frosst, and G. E. Hinton. "Dynamic Routing Between Capsules". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. ***30***. Curran Associates, Inc., 2017. DOI: 10.48550/arXiv.1710.09829.

[28] V. Mazzia, F. Salvetti, and M. Chiaberge, (2021) *"Efficient-capsnet: Capsule network with self-attention routing"* **Scientific reports *11***(1): 14634. DOI: 10.1038/s41598-021-93977-0.

[29] J. Rajasegaran, V. Jayasundara, S. Jayasekara, H. Jayasekara, S. Seneviratne, and R. Rodrigo. "DeepCaps: Going Deeper With Capsule Networks". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019. DOI: 10.1109/CVPR.2019.01098.

[30] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. 2017. DOI: 10.48550/arXiv.1704.04861. arXiv: 1704.04861 [cs.CV].

[31] A. Kirillov, Y. Wu, K. He, and R. Girshick. "PointRend: Image Segmentation As Rendering". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020. DOI: 10.1109/CVPR42600.2020.00982.

[32] L. Reiher, B. Lampe, and L. Eckstein. "A Sim2Real Deep Learning Approach for the Transformation of Images from Multiple Vehicle-Mounted Cameras to a Semantically Segmented Image in Bird's Eye View". In: *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. 2020, 1–7. DOI: 10.1109/ITSC45102.2020.9294462.

[33] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang. "DenseASPP for Semantic Segmentation in Street Scenes". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. DOI: 10.1109/CVPR.2018.00388.

[34] T. Takikawa, D. Acuna, V. Jampani, and S. Fidler. "Gated-SCNN: Gated Shape CNNs for Semantic Segmentation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019. DOI: 10.1109/ICCV.2019.00533.

[35] J. Lee, D. Kim, J. Ponce, and B. Ham. "SFNet: Learning Object-Aware Semantic Correspondence". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019. DOI: 10.1109/CVPR.2019.00238.

[36] R. P. K. Poudel, S. Liwicki, and R. Cipolla. *Fast-SCNN: Fast Semantic Segmentation Network*. 2019. DOI: 10.48550/arXiv.1902.04502. arXiv: 1902.04502 [cs.CV].

[37] E. Romera, J. M. Álvarez, L. M. Bergasa, and R. Arroyo, (2018) *"ERFNet: Efficient Residual Factorized ConvNet for Real-Time Semantic Segmentation"* **IEEE Transactions on Intelligent Transportation Systems *19***(1): 263–272. DOI: 10.1109/TITS.2017.2750080.

[38] Z. Guo, Z. Chen, T. Yu, J. Chen, and S. Liu. "Progressive Image Inpainting with Full-Resolution Residual Network". In: *Proceedings of the 27th ACM International Conference on Multimedia*. MM '19. Nice, France: Association for Computing Machinery, 2019, 2496–2504. DOI: 10.1145/3343031.3351022.

[39] S. Wang, L. Yi, Q. Chen, Z. Meng, H. Dong, and Z. He. "Edge-aware Fully Convolutional Network with CRF-RNN Layer for Hippocampus Segmentation". In: *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*. 2019, 803–806. DOI: 10.1109/ITAIC.2019.8785801.

[40] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. "Pyramid scene parsing network". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, 2881–2890. DOI: 10.48550/arXiv.1612.01105.

[41] H. Wang, X. Jiang, H. Ren, Y. Hu, and S. Bai. "SwiftNet: Real-Time Video Object Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, 1296–1305. DOI: 10.1109/CVPR46437.2021.00135.

[42] H. Li, P. Xiong, H. Fan, and J. Sun. "DFANet: Deep Feature Aggregation for Real-Time Semantic Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019. DOI: 10.1109/CVPR.2019.00975.

[43] S. Yan, C. Wu, L. Wang, F. Xu, L. An, K. Guo, and Y. Liu. "DDRNet: Depth Map Denoising and Refinement for Consumer Depth Cameras Using Cascaded CNNs". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018. DOI: 10.1007/978-3-030-01249-6_10.