Context-Integrated and Feature-Refined Network for Lightweight Object Parsing

Bin Jiang, Wenxuan Tu, Chao Yang, and Junsong Yuan, *Senior Member, IEEE*

***Abstract*— Semantic segmentation for lightweight object parsing is a very challenging task, because both accuracy and efficiency (e.g., execution speed, memory footprint or computational complexity) should all be taken into account. However, most previous works pay too much attention to one-sided perspective, either accuracy or speed, and ignore others, which poses a great limitation to actual demands of intelligent devices. To tackle this dilemma, we propose a novel lightweight architecture named Context-Integrated and Feature-Refined Network (CIFReNet). The core components of CIFReNet are the Long-skip Refinement Module (LRM) and the Multi-scale Context Integration Module (MCIM). The LRM is designed to ease the propagation of spatial information between low-level and high-level stages. Furthermore, channel attention mechanism is introduced into the process of long-skip learning to boost the quality of low-level feature refinement. Meanwhile, the MCIM consists of three cascaded Dense Semantic Pyramid (DSP) blocks with image-level features, which is presented to encode multiple context information and enlarge the field of view. Specifically, the proposed DSP block exploits a dense feature sampling strategy to enhance the information representations without significantly increasing the computation cost. Comprehensive experiments are conducted on three benchmark datasets for object parsing including Cityscapes, CamVid, and Helen. As indicated, the proposed method reaches a better trade-off between accuracy and efficiency compared with the other state-of-the-art methods.**

***Index Terms*— Object parsing, semantic segmentation, model efficiency, feature refinement, multi-scale context information.**

I. Introduction

SEMANTIC segmentation aims at labeling pixel-level signals associated with category, location, and shape for objects, which can be utilized in many applications such as robotic system, medical imaging, and object parsing [1]–[3]. Object parsing is supposed to segment the whole image into different semantic parts such as a building, a car, and a person, as shown in Fig. 1. The core challenge of object parsing is how to keep both high accuracy and real-time speed under resource-constrained environments. Hence for methods to be practically applicable, it is necessary that they have to be accurate, fast as well as resource-saving.

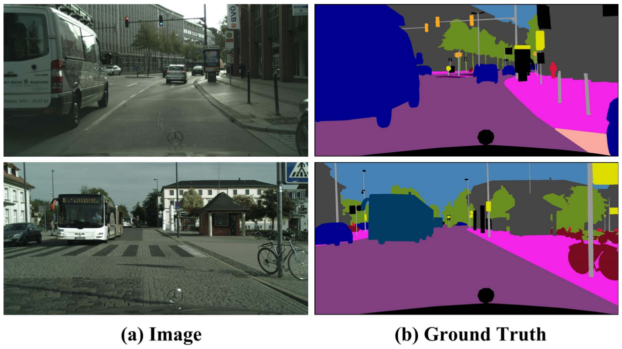


Fig. 1. Illustration of some urban scenes in Cityscapes dataset. An intelligent device must comprehensively implement scene interactivity with the complex environment, then make the driving decision timely.

Early Convolutional Neural Networks (CNNs) based works handle the object parsing task by designing U-shape [4]–[6] or Multi-scale [7]–[10] architectures, which can make full use of spatial details or context information for achieving high accuracy, as shown in Fig. 2(a) and Fig. 2(b). However, these methods rely on more convolutions and sophisticated operations, which are time-consuming and require enormous computation resources. For example, although the process of details recovery could benefit from the skip learning strategy in U-shape architecture, each layer in encoder transfers an equivalent number of feature maps to corresponding layer in decoder, leading to large amounts of extra computations [3]–[6]. Moreover, encoding multi-scale context information on the tail of the ResNet101-based network could deal with object variation cases and boost the recognition performance, but at the expense of calculating 2,048 feature maps by 3 × 3 regular convolution before feature integration [8]–[10]. To overcome these drawbacks, a series of works have tried to design lightweight models to achieve a real-time speed [11]–[13]. As illustrated in Fig. 2(c), a typical asymmetric encoder-decoder structure focuses on reducing the number of parameters for acceleration. Nevertheless, most of them heavily compromise accuracy to speed by compressing feature channels, resulting in the final MIoU (Mean Intersection over Union) score notably drops to 60% or even lower [12], [13].

Manuscript received May 27, 2019; revised December 22, 2019; accepted February 15, 2020. Date of publication March 11, 2020; date of current version March 19, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61702176 and Grant 61772191 and in part by the Hunan Provincial Natural Science Foundation of China under Grant 2017JJ3038. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ghassan Alregib. *(Corresponding author: Bin Jiang.)*

Bin Jiang, Wenxuan Tu, and Chao Yang are with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China (e-mail: jiangbin@hnu.edu.cn; twx@hnu.edu.cn; yangchaoedu@hnu.edu.cn).

Junsong Yuan is with the Department of Computer Science and Engineering, The State University of New York at Buffalo, Buffalo, NY 14260-2500 USA (e-mail: jsyuan@buffalo.edu).

Digital Object Identifier 10.1109/TIP.2020.2978583

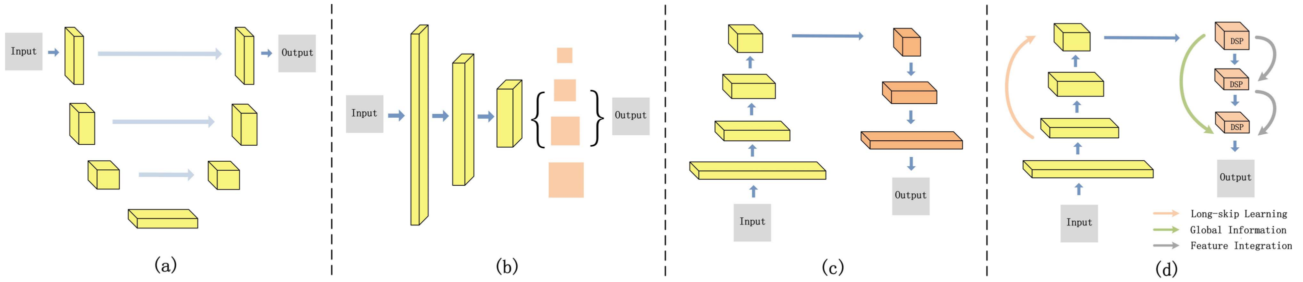


Fig. 2. Architecture comparison. From left to right: (a) U-shape structure. (b) Multi-scale context structure. (c) Asymmetric encoder-decoder structure. (d) The proposed CIFReNet.

Recently, there has been increasing interest in jointly considering a good trade-off between accuracy and speed [14]– [18]. Though these state-of-the-art methods could generate accurate results and maintain a real-time speed, we argue that most of them are still limited in terms of memory footprint or computational complexity. For instance, ERFNet [16] employs factorized convolutions with different dilation rates in encoder to maintain the accuracy-speed balance. Unfortunately, multiple regular transposed convolution layers in decoder may suffer from amounts of invalid floating point operations. Besides, ERFNet only handles single scale cases due to the fixed-size field of view in each layer, which ignores the significance of multi-scale context information and spatial details. In order to address these issues, another strategy [17] follows the principle of multi-branch framework to process multi-resolution inputs, then gathers them by a feature integration module. However, the additional branches based on regular CNNs bring large amounts of redundant parameters. Therefore, it is important to design a network that is able to perform high segmentation accuracy, and simultaneously process signals at a real-time speed under resource-constrained environments.

This motivates us to propose a lightweight architecture called Context-Integrated and Feature-Refined Network (CIFReNet), which is specifically tailored for resource constrained environments, as seen in Fig. 2(d). CIFReNet mainly includes two components: Long-skip Refinement Module (LRM) and Multi-scale Context Integration Module (MCIM). In pursuit of better accuracy, we firstly establish a single long-skip connection between shallow layers and deep layers to ease the propagation of low-frequency information. Then we apply channel attention mechanism to narrow the gap between multi-level information, which could adaptively revise some crucial details for feature refinement. After that, the Dense Semantic Pyramid (DSP) blocks, as the basic units of MCIM, are elaborately designed to aggregate multi-perspective context information together at various dilation rates. Meanwhile, the global prior knowledge is added into the MCIM to enlarge the field of view. Thus, MCIM learns the joint information of both local and global context, which guides the learning process more precisely. In order to improve the model efficiency, CIRFeNet employs the modified MobileNet V2 as encoder that contains 17 depthwise separate convolution layers [19]. The maximum number of each layer is no more than 320, which maintains the computation efficiency and guarantees adequate information during propagation simultaneously. To further reduce redundant parameters and invalid floating point operations, the depth-wise separable convolution [20] and the group convolution [21] are utilized to optimize the LRM and the MCIM. Note that DSP blocks are stacked in cascade rather than in parallel, refraining from large amounts of channel calculations after feature integration. Finally, the prediction maps are bilinearly up-sampled without any additional branch or transposed convolution layer, thus improving the resource utilization ratio. We demonstrate that the proposed CIFReNet obtains 70.9% MIoU on Cityscapes test set, 64.5% MIoU on CamVid test set, and 71.3% MIoU on Helen test set with less than 1.9 M parameters and only 7.3 GFLOPs. Meanwhile, it processes an image of 640 × 360 resolution at a speed of 62.5 FPS on a single NVIDIA GTX 1080Ti card. Experimental results demonstrate that our method reaches a better trade-off among overall performance compared with some state-of-the-art ones.

The key contributions of this paper are three-fold:

- A lightweight architecture named CIFReNet is proposed for object parsing. Compared with some state-of-the-art methods, CIFReNet obtains a better trade-off between accuracy and efficiency (e.g., execution speed, memory footprint, or computational complexity). As a side contribution, we will release our source code.[1]
- A slight yet effective LRM is designed, which adopts a long-skip connection with channel attention mechanism to provide a highway and a proper guidance for spatial information learning. Therefore, it can adaptively refine segmentation results in a coarse level for better accuracy.
- An efficient and powerful MCIM with cascaded DSP blocks is presented to capture multi-perspective context information and expand the field of view. In particular, the DSP block can encode much denser semantic information with an acceptable cost.

The remainder of this paper is organized as follows. Section II reviews related works in terms of object parsing tasks based on sematic segmentation. Section III presents the model design and each component of CIFReNet. Section IV conducts experiments and discusses the results. Finally, section V draws a conclusion.

[1] https://github.com/WxTu/CIFReNet

II. Related Works

Some recent works based on Fully Convolution Networks (FCNs) [22] have achieved promising results on public benchmarks [23], [24]. We then review the latest deep-learning-based methods for object parsing from lightweight-oriented and accuracy-oriented aspects.

A. Lightweight-Oriented Approaches

Lightweight semantic segmentation methods aim at solving the problem of slow speed and resource constraints on intelligent devices. These works can roughly be grouped into two categories: the speed-fast structure [11]–[13] and the accuracy-speed trade-off structure [14]–[18]. As for the former, ENet [12] adopts the combination of an initial block and a group of factorized filters for acceleration. Further, ESPNet [13] assembles Efficient Spatial Pyramid (ESP) modules into an encoder-decoder structure, outperforming ENet in terms of accuracy and speed. Although efforts have been made on designing an extremely speed-fast and resource-saving model, most of these methods sacrifice too much accuracy, so they can not generate sufficient precise information. The latter category tries to balance accuracy and speed, which has become an active research area in the last few years. Specifically, ERFNet [16] designs a convolutional factorization technique with dilation rate to make parameters less redundant and obtain the acceptable accuracy. Besides, ICNet [17] designs multi-branch cascaded sub-networks to achieve a fast speed, and then applies multi-scale resolution images as inputs for coarse-to-fine inference. Despite their success, so far most of them have taken less consideration on either space constraints or computation requirements, which is unfeasible for intelligent devices that require low memory footprint and computational complexity. Different from the aforementioned works, the proposed CIFReNet enhances the model efficiency by designing lightweight yet powerful LRM and MCIM, which boosts the spatial information learning and encodes denser multiple context information respectively.

B. Accuracy-Oriented Approaches

Towards high-quality results, most accuracy-oriented methods are designed to encode more spatial details and multi-scale context information.

1) Feature Refinement: Refining spatial information is a common challenge in semantic segmentation methods [3]–[6], [25], [26], which plays an important role in predicting the pixel-level localization. A direct solution is to design a symmetrical encoder-decoder model. For example, U-Net [3] adopts long-skip connections to refine details by fusing the hierarchical features of the backbone. Similarly, SegNet [4] adopts a typical U-Shape structure and utilizes the saved pooling indices to gradually compensate the spatial information for high-level features. U-shape design has also been presented in [5] and [6], the authors create an up-sampling stage corresponding to each down-sampling one, and fuse them in the decoder step by step. However, most of them directly utilize element-wise sum or channel concatenation to bridge the gap among multi-level features, resulting in both high computation burden and less efficiency of feature representations. In contrast, we adopt a simple long-skip residual learning module to keep low-frequency information easier bypassed from bottom to top. Meanwhile, we introduce more semantic information to guide the low-frequency features learning, which effectively refines the final prediction with slight computation increase.

2) Multi-Scale Context Integration: Multiple context information is generally regarded as a key factor to provide a good descriptor in object parsing works [7]–[10], [27]–[31]. PSPNet has exhibited the impressive performance by designing Spatial Pyramid Pooling (SPP) module for capturing abundant context information [7]. DeepLab V2 integrates multi-scale context information by the proposed Atrous Spatial Pyramid Pooling (ASPP) module with diverse dilation rates [8]. Following the same strategy, Chen *et al.* [9] feed global features into local context information to obtain larger field of view. Further, Ding *et al.* [27] manage to selectively integrate multi-scale features for each spatial position by a scheme of gated sum. Subsequently, Zhang *et al.* [28] present a scale-adaptive convolution to exploit long-range context information, which acquires a flexible size of receptive field. Although these methods perform well in solving the challenge of scene variations, most of them prefer heavy backbones and complicated modules to pursue high accuracy. These components lead to the time-consuming inference and bring much heavy overheads for object parsing. In order to reduce the computation burden while maintaining high accuracy, we propose the MCIM to aggregate both local and global context information with negligible computation cost.

III. Context-Integrated and Feature-Refined Network

A. Overview

In this section, we introduce a single-shot architecture called CIFReNet for lightweight object parsing, which mainly consists of the Long-skip Refinement Module (LRM) and the Multi-scale Context Integration Module (MCIM). CIFReNet aims to achieve an overall trade-off in terms of accuracy and efficiency (e.g., execution speed, memory footprint, or computational complexity) by efficiently learning spatial and contextual information.

As depicted in Fig. 3, given an input, we firstly feed it into our backbone network to obtain the semantic features. The Output Stride (OS) of the encoder is reasonably set to 8 for high-resolution datesets (e.g., Cityscapes [23]) and set to 4 for low-resolution datesets (e.g., CamVid [32] and Helen [33]), so as to save the memory resources and preserve more spatial details during the training process. Furthermore, we replace the last four sub-sampling operations with dilation convolutions, and apply a group of hybrid dilation rates {2, 3, 5, 7} to maintain the field of view, as illustrated in Table I. Note that all initial settings mentioned above are determined according to ablation studies.

Next, we perform a Long-skip Refinement Module, as shown in the red box of Fig. 3. For the first step, we establish a simple long-skip connection between spatial

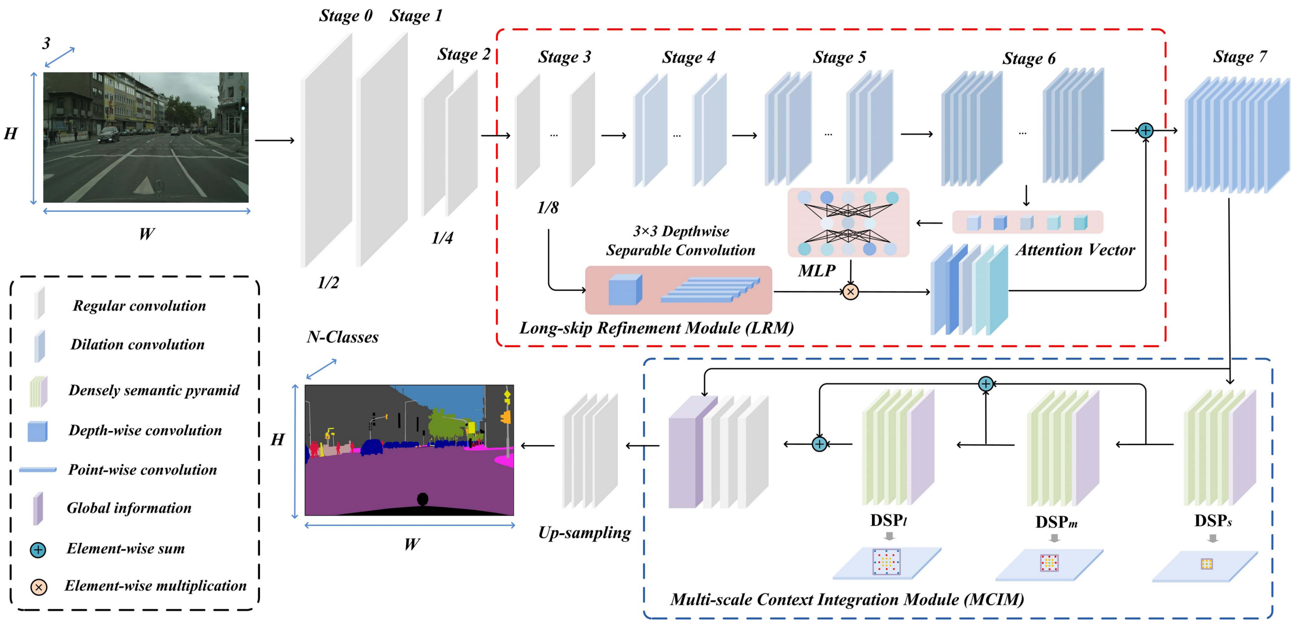


Fig. 3. An overview of Context-Integrated and Feature-Refined Network (CIFReNet). The dotted red box and blue box represent the Long-skip Refinement Module (LRM) and the Multi-scale Context Integration Module (MCIM), respectively.

TABLE I

NETWORK ARCHITECTURE. C_o: OUTPUT CHANNELS. N: THE NUMBER OF CATEGORIES. LRM: LONG-SKIP REFINEMENT MODULE. MCIM: MULTI-SCALE CONTEXT INTEGRATION MODULE. UL: UP-SAMPLING LAYER

Model	Type	Dilation	C_o	Repeat
Stage$_0$	Inverted Residual Block	1	32	1
Stage$_1$	Inverted Residual Block	1	16	1
Stage$_2$	Inverted Residual Block	1	24	2
Stage$_3$	Inverted Residual Block	1	32	3
LRM	-	-	160	1
Stage$_4$	Inverted Residual Block	2	64	4
Stage$_5$	Inverted Residual Block	3	96	3
Stage$_6$	Inverted Residual Block	5	160	3
Stage$_7$	Inverted Residual Block	7	320	1
MCIM	-	-	400	1
UL	Up-sampling Layer	-	N	1

layer and semantic layer. The high-dimensional spatial features are generated by a 3 × 3 depth-wise separate convolution. For the second step, the high-level features are transformed into a group of weight vectors and used to refine the high-dimensional spatial features. Through the above process, we combine the refined features with original high-level features by an element-wise sum operation. Such a design eases the propagation of low-frequency information and boosts the quality of feature refinement for better segmentation results. Moreover, LRM contains only one slight long-skip connection with negligible overheads.

Thereafter, we feed semantic features outputted from the tail of the backbone into Multi-scale Context Integration Module to gather both local and global context information. As shown in the blue box of Fig. 3, three lightweight DSP blocks with a global constraint are piled up in the network. Such a cascaded design has two advantages. On the one hand, it can enlarge the field of view by increasing the depth of network. On the other hand, it follows the principle of "deep and thin" [34], which considerably boosts the model efficiency. Additionally, the semantic features outputted from each DSP block are integrated together, which allows the network to jointly learn context information at multi-level scales for better recognition performance.

Finally, we directly up-sample the output feature maps back to input size by Up-sampling Layer (UL). In the following sections, we will elaborate the design of LRM and MCIM in detail. Before that, we summarize all notations in Table II.

B. Long-Skip Refinement Module (LRM)

As discussed in Section I, the U-shape architecture illustrates that skip learning is beneficial to model performance improvement, but suffers from low speed and double resource utilization. Inspired by the success of long-skip learning [35] and channel attention mechanism [36], we propose the Long-Skip Refinement Module to tackle the above issue for better segmentation performance by effectively and efficiently learning spatial information.

1) Basic Long-Skip Learning: Deeper stage provides more semantic information but loses too many visual details. Although shallower stage preserves more spatial details, it contains much irrelevant noise. Based on this observation, we firstly manage to establish a long-skip connection to directly integrate multi-level features between shallow and deep stages. Extensive ablation studies have demonstrated that adopting a long-skip learning manner from Stage$_3$ to Stage$_6$ delivers a better trade-off in terms of overall performance compared with other options, which contains ten inverted residual blocks (as illustrated in Table I) in total.

As depicted in the red box of Fig. 3, given shallow feature maps (generated by Stage$_3$) as an input

TABLE II
BASIC NOTATIONS FOR THE PROPOSED METHOD

Notations	Meaning
$H \times W$	The size of feature maps
C_i	The number of input channels
C_o	The number of output channels
F_s	The low-dimensional shallow feature maps
$F^{'}{}_s$	The high-dimensional shallow feature maps
F_a	The high-dimensional abstract feature maps
V	The set of attention vectors for $F^{'}_s$
$V^{'}$	The set of dimension-reduced attention vectors
$I^{H \times W \times C_i}$	The input of DSP
$I^{'\,H \times W \times (C_o \times r)}$	The channel-reduced feature maps of DSP
D	The set of dilation rates in DSP
L	The set of n-groups features in DSP
$G^{1 \times 1 \times (C_o \times r)}$	The global feature vectors in DSP
$G^{'\,H \times W \times (C_o \times r)}$	The global context in DSP
$O^{H \times W \times C_o}$	The output of DSP
$O^{'\,H \times W \times C_o}$	The residual output of DSP
$O^{''\,H \times W \times C_o}$	The output of the integrated semantics in MCIM
$F^{H \times W \times C_i}_{Stage_7}$	The feature maps from the tail of the encoder
$G^{''\,H \times W \times C}$	The global context generated by $F^{H \times W \times C_i}_{Stage_7}$
$Y^{''\,H \times W \times (C_o + C)}$	The output of MCIM

$F_s = \{f^{H \times W}_{s_1}, f^{H \times W}_{s_2}, \cdots, f^{H \times W}_{s_{C_i}}\}$, we firstly apply a depth-wise separable convolution to transform low-dimensional F_s into high-dimensional $F'_s = \{f'^{H \times W}_{s_1}, f'^{H \times W}_{s_2}, \cdots, f'^{H \times W}_{s_{C_o}}\}$, which is beneficial for feature representations but not computation expensive, as formulated in Eq. (1). Then we fuse F'_s and abstract features (generated by Stage$_6$) $F_a = \{f^{H \times W}_{a_1}, f^{H \times W}_{a_2}, \cdots, f^{H \times W}_{a_{C_o}}\}$ by element-wise sum, which serves as the basic long-skip structure for residual learning. By this way, more abundant low-frequency information can be bypassed conveniently in the network.

$$F'_s = \delta(pw^{1 \times 1} \times \delta(dw^{3 \times 3} \times F_s + b) + b), \tag{1}$$

where $dw^{3 \times 3}$ and $pw^{1 \times 1}$ are defined as a 3 × 3 depth-wise convolution and a 1 × 1 point-wise convolution, respectively. b represents the bias vector. $\delta(\cdot)$ indicates the operations of both Batch Normalization (BN) [37] and Parametric Rectified Linear Unit (PReLU) [38] function.

2) Spatial Feature Refinement: Though Zhang *et al.* [35] have proven that long-skip connection not only eases the optimization process but also promotes network learning in a coarse level, we argue that straightly embedding shallow features along with much noise into valuable semantic features is meaningless. To tackle this dilemma, we utilize the high-level features to provide a constraint for low-level features learning rather than simply performing element-wise sum or concatenation. Specifically, we reshape the high-level feature maps F_a into feature vectors $V = \{v^{1 \times 1}_1, v^{1 \times 1}_2, \cdots, v^{1 \times 1}_{C_o}\}$ via a global average pooling layer. Then we feed V into Multi-Layer Perception (MLP) layers to obtain dimension-reduced feature vectors $V' = \{v'^{1 \times 1}_1, v'^{1 \times 1}_2, \cdots, v'^{1 \times 1}_{C_o}\}$, similar to SENet [36]. Finally, a softmax layer is applied to calculate the attention value which allows F'_s to adaptively adjust its selection. We present this process as follows:

$$v_k = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} f_{a_k}(i, j), \tag{2}$$

$$y^{H \times W}_k = \frac{exp(v'_k)}{\sum_{i=1}^{C_o} exp(v'_k)} \otimes f'^{H \times W}_{s_k} \oplus f^{H \times W}_{a_k}, \tag{3}$$

where $y^{H \times W}_k$ refers to the k-th refined output feature map, $k \in \{1, 2, \cdots, C_o\}$. $\otimes$ and $\oplus$ refer to element-wise multiplication and element-wise sum, respectively.

C. Multi-Scale Context Integration Module (MCIM)

In the task of object parsing, most current methods design multi-scale modules such as PPM [7] and ASPP [8] to effectively deal with object variation cases, but heavily decrease the model efficiency. To overcome this limitation, the Multi-scale Context Integration Module that consists of three DSP blocks and global information is established, as shown in blue box of Fig. 3. The MCIM has only 0.12 M parameters and requires fewer resource utilization (0.49 MB and 2.48 GFLOPs). Next, we will further discuss about both efficiency and effectiveness of MCIM.

1) Dense Semantic Pyramid (DSP): The Dense Semantic Pyramid block, with less than 0.03 M parameters, is the core component of MCIM. We then reveal the design of DSP block in detail as the following four steps.

a) Lightweight convolutional techniques: As we all know, depth-wise separable convolution [20] and group convolution [21] have been proven more efficient in terms of memory footprint and computational complexity while keeping similar accuracy compared with the regular convolution. Therefore, we take advantages of them to make the DSP block more computationally efficient. As displayed in Fig. 4, given feature maps $I^{H \times W \times C_i}$ as an input of the DSP block, we firstly feed it into a 1 × 1 group point-wise convolution layer to obtain channel-reduced feature maps $I'^{H \times W \times (C_o \times r)}$, where r refers to channel reduction ratio. Then $I'^{H \times W \times (C_o \times r)}$ are successively sent to n parallel depth-wise separable convolutions followed by BN and PReLU, in order to generate n groups feature maps $L = \{l^{H \times W \times (C_o \times r), d}_1, l^{H \times W \times (C_o \times r), d}_2, \cdots, l^{H \times W \times (C_o \times r), d}_n\}$, where d denotes the dilation rate. With these lightweight convolutional techniques [20], [21], the dense convolutional filters among all channels are uniformly changed to be sparse, thus decreasing the computation costs and memory requirements. As listed in Table III, the results of bottom two rows clearly show that the proposed DSP block achieves about 9 times reduction in parameters compared with the one based on regular convolutions.

b) Denser dilation sampling: Thereafter, all neurons in each branch share the same field of view at a single scale, which is hard to deal with scale variation cases for object parsing. Different from the previous sparse feature sampling strategy [8], [9], we incorporate contextual information at multiple dense scales, because pixels near the target usually contain more useful semantic information. To gather the local context information as much as possible, we successively replace original dilation rates in 3 × 3 depth-wise

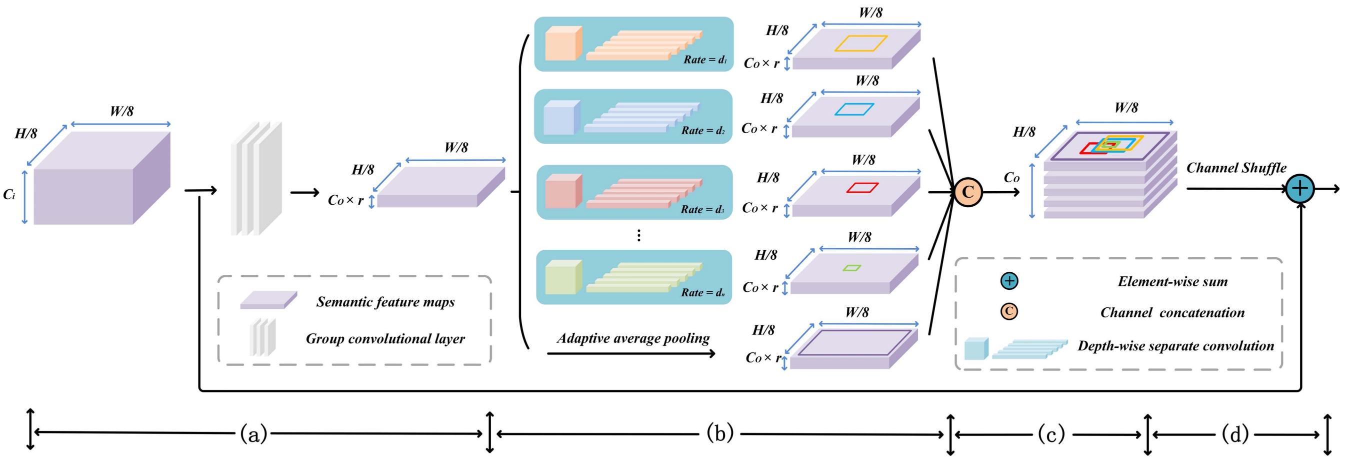


Fig. 4. An overview of the Dense Semantic Pyramid (DSP) block. (a) Channel reduction. (b) Dense dilation sampling. (c) Semantic features integration. (d) Channel shuffle operation.

TABLE III

DIFFERENT TYPES OF CONVOLUTIONS AND MODULES FOR COMPARISON. ♯PARAMS: THE NUMBER OF PARAMETERS. m^2 IS THE KERNEL SIZE. g IS THE NUMBER OF GROUPS. n IS THE NUMBER OF PATH. RC: REGULAR CONVOLUTION. GC: GROUP CONVOLUTION. DSC: DEPTH-WISE SEPARABLE CONVOLUTION. RC-DSP: REGULAR CONVOLUTION IN DSP. ASSUME THAT $C_i = 320$, $C_o = 320$, $r = 1/4$, $m = 3$, $g = 4$ AND $n = 4$

Type	♯Params(M)
RC	$C_i \times C_o \times m^2 = 0.92$
GC	$C_i \times C_o \times m^2/g = 0.23$
DSC	$C_i \times m^2 + C_i \times C_o = 0.11$
RC-DSP	$C_i \times (C_o \times r) + (C_o \times \mathrm{r})^2 \times m^2 \times n = 0.26$
DSP	$C_i \times (C_o \times r)/g + ((C_o \times r) \times m^2 + (C_o \times r)^2) \times n = 0.03$

convolutions with a group of coprime dilation rates $D = \{d_1, d_2, \cdots, d_n\}$ (inspired by the prior work [39]). As depicted in Fig. 5(b), a much denser feature sampling measure is provided to capture more relevant sub-regions $L = \{l_1^{H \times W \times (C_o \times r), d_1}, l_2^{H \times W \times (C_o \times r), d_2}, \cdots, l_n^{H \times W \times (C_o \times r), d_n}\}$ compared with the case in Fig. 5(a). In this way, the DSP block could enrich feature representations without any extra parameters. Although channel pruning operation dramatically decreases the computation cost, the difficulty in too much information loss becomes a major issue. To alleviate the performance degradation, we reasonably set r to 0.2 and n to 4 by performing ablation studies. The above process can be formulated as:

$$I'^{H \times W \times (C_o \times r)} = \delta(w^{1 \times 1} \times I^{H \times W \times C_i} + b), \qquad (4)$$

$$l_k^{H \times W \times (C_o \times r), d_k} = DSC_{d_k}(I'^{H \times W \times (C_o \times r)}), \qquad (5)$$

where $l_k^{H \times W \times (C_o \times r), d_k}$ refers to the k-th path feature maps in the DSP block, $k \in \{1, 2, \cdots, n\}$. $DSC_{d_k}(\cdot)$ denotes the Depthwise Separable Convolution with dilation rate d_k.

c) Local and global context extraction: The above design of DSP block is able to perceive patches or pixels locally, but falls short of a global view, especially for large objects and confusion categories. To remedy this defect, we add a path with small computation overheads, and then generate the global vectors $G^{1 \times 1 \times (C_o \times r)}$ from feature maps $I'^{H \times W \times (C_o \times r)}$ by adopting a global average pooling (GAP) layer, similar to [9]. Specifically, GAP performs down-sampling operation by computing the mean of the height H and width W dimensions of the input. Subsequently, the up-sampling layer utilizes the weighted average of two translated pixel values for each output pixel value of $G'^{H \times W \times (C_o \times r)}$ by bilinear interpolation. After that, the features are integrated from all parallel paths. Please note that directly integrating them stemmed from a group convolution layer may weaken feature representations [21]. Consequently, the channel shuffling operation is applied to ease cross-group information $O^{H \times W \times C_o}$, as formulated below:

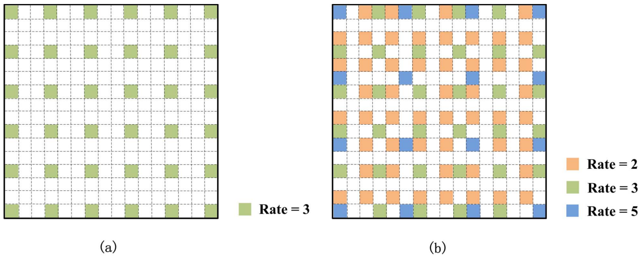


Fig. 5. Dilation sampling comparison. (a) Sparse Feature Sampling. (b) Dense Feature Sampling.

$$G^{1 \times 1 \times (C_o \times r)} = GAP(I'^{H \times W \times (C_o \times r)}), \qquad (6)$$

$$G'^{H \times W \times (C_o \times r)} = U(G^{1 \times 1 \times (C_o \times r)}), \qquad (7)$$

$$O^{H \times W \times C_o} = S(C[L^{H \times W \times (C_o \times r) \times n}, G'^{H \times W \times (C_o \times r)}]), \qquad (8)$$

where $GAP(\cdot)$ denotes a global average pooling layer, $U(\cdot)$ represents the bilinear interpolation operation. $C[\cdots]$ means the channel concatenation, and $S(\cdot)$ corresponds to the channel shuffling operation.

d) Short-skip residual learning: To promote the flows of contextual information throughout the network, we adopt a shortcut connection that strengthens gradient back-propagation in the DSP block, inspired by the principle of ResNet [40]. Formally, the combination of input $I^{H\times W\times C_i}$ and non-linear transformation output $O^{H\times W\times C_o}$ is described as:

$$O'^{H\times W\times C_o} = I^{H\times W\times C_i} \oplus O^{H\times W\times C_o}\ s.t.\ C_i \equiv C_o. \quad (9)$$

2) Going Deeper With Global View: In contrast to previous multi-scale networks [7]–[10] that calculate large number of feature maps in a parallel-branch module, we apply three DSP blocks in cascade rather than in parallel to save computation resources and adopt image-level features in MCIM to enlarge the field of view, as illustrated in Fig. 3.

a) Multi-scale object variations: We follow the strategy of pixel sampling rates in ASPP but keep dilation rates denser in the DSP block. Fig. 3 presents that each DSP block has its own function for object parsing. Specifically, the first DSP block with a group of hybrid dilation rates $D_s = \{1, 2, 3, 5\}$ mainly focuses on small size objects (e.g., traffic signs). On the basis of the former, we employ the second DSP block with a group of hybrid dilation rates $D_m = \{7, 9, 11, 13\}$ to focus on medium size objects (e.g., cars). Further, the last DSP block with a group of hybrid dilation rates $D_l = \{17, 19, 21, 23\}$ is designed to focus on large size objects (e.g., buildings). It is worth mentioning that since high-level feature maps contain a limited number of channels and the resolution of feature maps is small, thus setting large dilation rate does not increase too much computation overheads. Finally, we successively combine the output of each DSP block by element-wise sum to jointly encode multi-level semantics for better recognition performance, similar to FCN [22].

b) Enlarging the field of view: As formulated in Eq. (10) and Eq. (11), each convolution layer with dilation rate D and kernel size K in the DSP block could obtain the field of view R. Therefore, stacking all DSP blocks together can theoretically obtain the largest field of view R_{max}:

$$R = (D-1)\times(K-1)+K, \quad (10)$$

$$R_{max} = R^{s}_{max} + R^{m}_{max} + R^{l}_{max} - 2, \quad (11)$$

for instance, the MCIM module will obtain the field of view with maximum size 83.

We argue that the size of effective field of view is practically smaller, since amounts of information (e.g., marginal areas of feature maps) is abandoned as the network goes deeper. To handle this problem, we add a GAP layer on the tail of the backbone to enlarge the field of view. The global information $G''^{H\times W\times C}$ is extracted from $F^{H\times W\times C_i}_{Stage_7}$, which keeps the valid field of view large enough from a macroscopic aspect. Then we combine $G''^{H\times W\times C}$ with integrated semantics $O'' = \{O'^{H\times W\times C_o}_1 \oplus O'^{H\times W\times C_o}_2 \oplus O'^{H\times W\times C_o}_3\}$ to generate the final output of MCIM. This process can be expressed as:

$$G''^{H\times W\times C} = U(GAP(\delta(w^{1\times 1}\times F^{H\times W\times C_i}_{Stage_7}+b))), \quad (12)$$

$$Y^{H\times W\times(C_o+C)} = C[O''^{H\times W\times C_o}, G''^{H\times W\times C}]. \quad (13)$$

IV. Experimental Evaluation

To evaluate the performance of CIFReNet, we detailedly present an experimental procedure as below.

A. Training Protocol

1) Datasets: **Cityscapes Dataset.** The Cityscapes is a popular dataset for urban object parsing [23]. It consists of 25,000 annotated 2,048 × 1,024 resolution images. The fine-annotated dataset contains 5000 images including 19 valid classes. There are 2,975 images for training, 500 images for validation, and 1,525 images for testing. Note that the ground truth of testing set is unavailable, we shall submit the results to an online test server. Due to the limited physical memory, we randomly sub-sample image resolution to 1,024 × 512 patches for training.

CamVid Dataset. The CamVid is another urban object parsing dataset [32] which contains 12 classes in total. It consists of 367 frames, 101 frames, and 223 frames for training, validation, and testing, respectively. The original frame resolution of CamVid is 960 × 720. Following the prior works [4], [12], we randomly sub-sample all images to 480 × 360 patches for training.

Helen Dataset. The Helen is a widely-used dataset for facial object parsing [33], which consists of 2,330 face images with 11 labeled categories including face, nose, left eye, right eye, left eyebrow, right eyebrow, upper lip, inner mouth, lower lip, hair and background. Helen dataset is divided into 2,000 images, 230 images, and 100 images for training, validation and, testing. We randomly rescale the image resolution to 512 × 512 patches for training.

2) Metrics: **Segmentation accuracy.** The Mean Intersection over Union (MIoU) is commonly adopted for semantic segmentation accuracy. Suppose n is the number of classes, we can compute it as:

$$MIoU = \frac{1}{n}\sum_{i=1}^{n}\frac{p_{ii}}{\sum_{j=1}^{n} p_{ij} + \sum_{j=1}^{n} p_{ji} - p_{ii}}, \quad (14)$$

where p_{ij} is the number of pixels that belong to i predicted to class j, p_{ii} refers to the true positive. p_{ij} and p_{ji} refer to the false positive and false negative, respectively.

Execution speed. The amount of forward pass time in millisecond (ms) that a network takes to process an image, which is generally measured by frames per second (FPS).

Network parameters. The total number of weights and biases of each layer in the network.

Memory footprint. The storage space that is required to store the network parameters.

Computational complexity. The number of float-point operations (FLOPs) for measuring how quickly and effectively a CNN model works.

3) Training Details: We implement all experiments in Pytorch with NVIDIA 1080Ti GPU cards. All ablation results are evaluated on Cityscapes validation set. To avoid the overfitting, we randomly scale images from 0.5 to 1.5 ratio and left-right flip them for all datasets. Besides, we employ the mean subtraction and add a random rotation operation from −3 to 3 degrees during training process. Following the prior protocol [7], we set initial learning rate to 0.005 and employ "Poly" learning rate policy by $1-(\frac{iter}{max\_iter})^{power}$ with a power 0.9. In the end-to-end learning, the network is trained by using Stochastic Gradient Descent (SGD) optimization

TABLE IV

ABLATION STUDIES ON BACKBONE CHOICES. ♯PARAMS: THE NUMBER OF PARAMETERS. ♯NS: NETWORK SIZE

Method	♯Params(M)	♯NS(MB)	MIoU(%)
FCN-VGGNet16	134.35	524.82	62.29
FCN-ResNet101	42.67	167.18	70.70
FCN-ResNet50	23.68	92.75	67.78
FCN-ResNet18	11.30	44.19	66.82
FCN-DenseNet121	7.09	28.11	68.41
FCN-MobileNet V2	1.93	7.68	67.66

TABLE V

ABLATION STUDIES ON INITIAL SETTINGS. OS: OUTPUT STRIDE

OS = 8	OS = 16	OS = 32	MIoU(%)
√			59.41
	√		61.39
		√	60.75

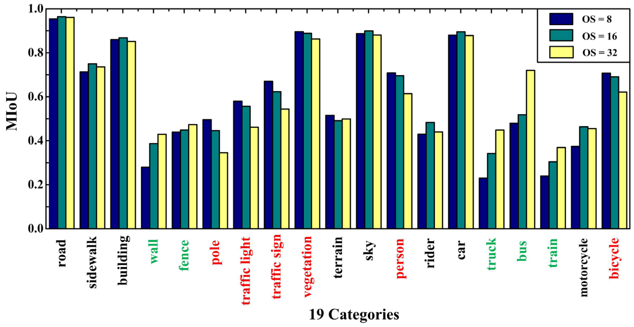


Fig. 6. The performance of 19 categories with different OS settings on Cityscapes validation set. Note that the categories in Red imply that setting smaller OS could boost the recognition of relatively small objects, and the categories in Green imply that setting larger OS could boost the recognition of relatively obvious ones.

TABLE VI

ABLATION STUDIES ON INITIAL SETTINGS. HDRS: HYBRID DILATION RATES STRATEGY. UL: UP-SAMPLING LAYER. DA: DATA AUGMENTATION. FLOPS ARE ESTIMATED ON A 3 × 640 × 360 INPUT

HDRS	UL	DA	FLOPs(G)	MIoU(%)
{1, 1, 1, 1}			6.89	59.41
{2, 2, 4, 4}			6.92 (0.03▲)	66.15 (6.74▲)
{1, 2, 3, 5}			6.91 (0.02▲)	66.17 (6.76▲)
{2, 3, 5, 7}			6.94 (0.05▲)	67.36 (7.95▲)
{3, 5, 7, 11}			6.97 (0.08▲)	69.41 (10.0▲)
{2, 3, 5, 7}		√	6.94 (0.05▲)	67.57 (8.16▲)
	√	√	6.95 (0.06▲)	67.84 (8.43▲)

algorithm, of which the momentum is 0.9 and weight decay is 5e-4. In addition, the pixel-wise cross-entropy error is employed as our loss function.

B. Modified MobileNet V2

1) Ablation for Backbone Choices: Since designing a novel backbone from scratch often requires expensive training resources, we investigate some pretrained backbone networks that are widely adopted in current segmentation models and make a comparison among them. For fairness, all experiments are carried out on a FCN-based model. As shown in Table IV, FCN-VGGNet16 is obviously not comparable to FCN-MobileNet V2 in both accuracy and efficiency due to the limited learning capacity and heavy structure of itself. Moreover, though FCN-ResNets achieve similar or better accuracy compared with FCN-MobileNet V2, the former generally contain enormous parameters and require more memory resources. Additionally, we observe that FCN-DenseNet121 is a pretty good choice, which yields a solid MIoU score of 68.41% while the requirement of resources is far less than FCN-ResNets. Further, FCN-DenseNet121 outperforms FCN-MobileNet V2 in terms of accuracy by 0.75% MIoU, but incurs about 4 times parameters and storage complexity increase. To sum up, all above comparisons demonstrate that employing MobileNet V2 as feature extractor could more efficiently bring the great benefit for object parsing.

2) Ablation for Output Stride: In this part, we conduct several experiments to explore an appropriate output stride (OS) value. Since processing high-resolution images (3 × 2,048 × 1,024) limits GPU resources, we do not consider further denser feature maps (e.g., OS $\leqslant$ 4). As seen in Table V and Fig. 6, employing OS $=$ 16 or 32 obtains a bit higher accuracy than the case with OS $=$ 8, especially for obvious objects marked with green color. It reflects that successive sub-sampling layers could bring sufficient field of view and make features more discriminative. Unfortunately, the performance of larger OS in small objects marked with red color are unsatisfied, resulting from amounts of information loss when setting OS to 16 or 32. Therefore, the value of OS is set to 8, so as not to make an irreversible affect for spatial details recovery during model inference.

3) Ablation for Hybrid Dilation Rates Strategy and Baseline: To evaluate the performance of dilation strategy in the backbone, we compare several proposals with the vanilla case where HDRS $= \{1, 1, 1, 1\}$. Note that the abbreviation HDRS refers to the hybrid dilation rates strategy applied in our backbone from Stage$_4$ to Stage$_7$. As summarized in Table. VI, we find out that ***a***) Compared with the original one, replacing the last four sub-sampling layers with dilation convolutions generally yields about 6%-10% MIoU boost, which illustrates the effectiveness of our strategy for backbone design. ***b***) Adopting HDRS $= \{2, 2, 4, 4\}$ indeed benefits the accuracy performance, but it is not the best choice. Due to same or geometric dilation ratios, the higher layer mostly samples information in the same region as the former one, thus making the field of view restricted. ***c***) The model obtains better segmentation results as the group of dilation rates becomes larger. However, the filters become too sparse to cover any relevant information, which requires more computation resources and leads to the aggravation of "Gridding Issue". The red box of Fig. 7 intuitively presents the comparison between HDRS $=$ $\{2, 3, 5, 7\}$ and HDRS $= \{3, 5, 7, 11\}$ about this issue. For further performance improvement, we suggest that employing

TABLE VII

PERFORMANCE COMPARISONS WITH VARIOUS LONG-SKIP MODULES. L_n: LOW-FREQUENCY INFORMATION OUTPUTTED FROM n-TH STAGE, $n \in (1, 2, 3)$. H_n: HIGH-FREQUENCY INFORMATION OUTPUTTED FROM n-TH STAGE, $n \in (6, 7)$. CA: CHANNEL ATTENTION. FPT: FORWARD PASS TIME. ♯PARAMS: THE NUMBER OF PARAMETERS. NOTE THAT FPT AND FLOPS ARE ESTIMATED ON A 3 × 640 × 360 INPUT

Model	L_1	L_2	L_3	H_6	H_7	CA	Sum	FPT(ms)	♯Params(M)	FLOPs(G)	MIoU(%)
Baseline								13.52	1.82	6.95	67.84
a	√			√		√	√	14.83 (1.31▲)	1.83 (0.01▲)	6.96 (0.01▲)	68.43 (0.59▲)
b		√		√		√	√	14.20 (0.68▲)	1.83 (0.01▲)	6.97 (0.02▲)	68.32 (0.48▲)
c			√	√		√	√	14.11 (0.59▲)	1.83 (0.01▲)	6.97 (0.02▲)	68.68 (0.84▲)
d			√		√	√	√	14.51 (0.99▲)	1.85 (0.03▲)	6.99 (0.04▲)	68.82 (0.98▲)
e			√	√			√	14.04 (0.52▲)	1.83 (0.01▲)	6.97 (0.02▲)	68.02 (0.18▲)

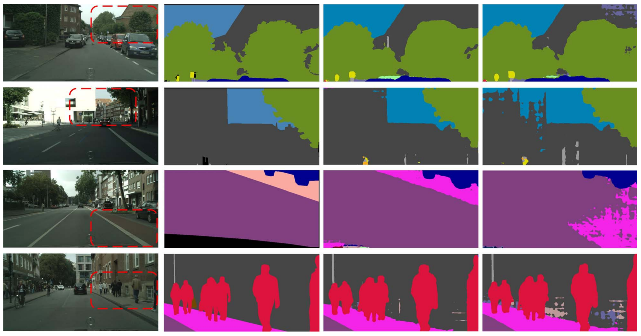

(a) Image (b) Ground Truth (c) {2, 3, 5, 7} (d) {3, 5, 7, 11}

Fig. 7. Visualization results of "Gridding Effect". From left to right: Image, Ground Truth, the group of hybrid dilation rates {2, 3, 5, 7} and {3, 5, 7, 11} added into the backbone.

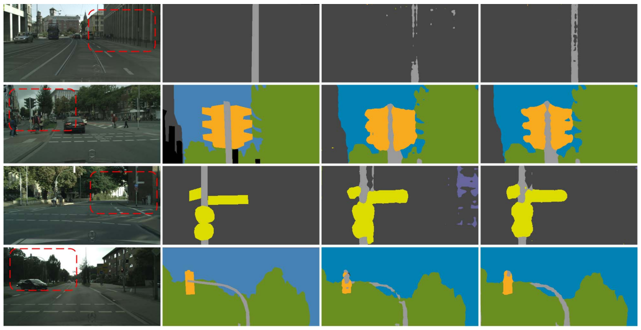

(a) Image (b) Ground Truth (c) Without LRM (d) With LRM

Fig. 8. Visualization results of LRM on Cityscapes validation set. From left to right: Image, Ground Truth, Without LRM, With LRM.

HDRS = {2, 3, 5, 7} to ensure both accuracy and efficiency. Additionally, instead of directly up-sampling the final feature maps by a simple convolution layer, we elaborately design an up-sampling layer as a transition operation for better information recovery. Further results show that the baseline model with up-sampling layer and data argumentation yields higher performance up to 67.84% MIoU, while the extra computation cost is negligible.

C. LRM and MCIM

Here we evaluate the performance of Long-skip Refinement Module and Multi-scale Context Integration Module.

1) Ablation for Long-Skip Refinement Module: As shown in Table VII, we carry out several ablation studies to reveal the effectiveness of LRM. Note that the check marks refer to the two stages united by a long-skip connection. We firstly fix the semantic feature layer (e.g., Stage$_6$) and then vary the shallow feature layer (e.g., Stage$_1$ to Stage$_3$), so as to evaluate the quality of various low-frequency features during the attention refinement process. As seen, Table VII has demonstrated that methods based on long-ship learning strategy all outperform the baseline method. Nevertheless, the performance of these methods are not similar due to their distinctive structures. Specifically, we observe that module *c* achieves comparable or slightly better performance in terms of both accuracy and efficiency than module *a* and module *b*. We attribute the superiority of module *c* as two aspects: ***a***) Different from module *a* and module *b* where shallow layers contain large amounts of noise in feature maps, module *c* could generate purer spatial information after deeper convolution layers, which is more effective for the attention refinement process and the later feature fusion. ***b***) Module *c* achieves a faster inference and avoids some needless computation cost without any extra down-sampling operation on high resolution feature maps. Moreover, although module *d* outperforms module *c* in terms of accuracy, we also observe that the former brings more than 2 times computation cost increase compared with the latter one, which is inconsistent with our purpose of a lightweight design. In addition, the accuracy of module *e* where a straightforward fusion of low-level and high-level features is implemented by element-wise sum yields only 0.18% MIoU boost. This indicates that such an manner may limitedly improve recognition performance, because an excessive semantic gap between low-frequency and high-frequency features helps little for spatial information learning. Based on all above observations, we regard module *c* as the proposed LRM, which achieves 0.84% MIoU improvement with 0.01 M extra parameters and 0.02 GFLOPs increase compared with the baseline method. Also, these results well demonstrates the effectiveness of LRM in terms of both accuracy and efficiency. The visual results comparison between the baseline method and the one with LRM is provided in Fig. 8 and Fig. 9, which also shows that some small objects marked with red color (e.g., poles, traffic signs) could benefit from the proposed LRM.

2) Ablation for Dense Semantic Pyramid Block: To investigate the effect on different settings of *n* and *r* for model performance, we conduct some experiments under the same

TABLE VIII

COMPARISONS OF CASCADED DSP BLOCKS WITH DIFFERENT SETTINGS WHEN ADOPTING MODIFIED MOBILENET V2 AS BACKBONE. n: THE NUMBER OF PATHS IN DSP. r: CHANNEL REDUCTION RATIO. D_k REPRESENTS SAMPLING SCALES OF DSP$_k$ BLOCK, $k \in (s, m, l)$. GAP: GLOBAL AVERAGE POOLING. ♯PARAMS: THE NUMBER OF PARAMETERS. NOTE THAT FLOPS ARE ESTIMATED ON A 320 × 256 × 128 INPUT

n	r	D_s	D_m	D_l	GAP	♯Params(M)	FLOPs(G)	MIoU(%)
2	1/2	$\{6_1, 6_2, \cdots, 6_n\}$	$\{12_1, 12_2, \cdots, 12_n\}$	$\{18_1, 18_2, \cdots, 18_n\}$		0.21	6.73	70.48
4	1/4	$\{6_1, 6_2, \cdots, 6_n\}$	$\{12_1, 12_2, \cdots, 12_n\}$	$\{18_1, 18_2, \cdots, 18_n\}$		0.11	3.58	70.89
8	1/8	$\{6_1, 6_2, \cdots, 6_n\}$	$\{12_1, 12_2, \cdots, 12_n\}$	$\{18_1, 18_2, \cdots, 18_n\}$		0.06	2.00	70.53
16	1/16	$\{6_1, 6_2, \cdots, 6_n\}$	$\{12_1, 12_2, \cdots, 12_n\}$	$\{18_1, 18_2, \cdots, 18_n\}$		0.04	1.21	70.23
4	1/4	{1, 2, 3, 5}	{7, 9, 11, 13}	{17, 19, 21, 23}		0.11	3.57	71.58
4	1/(4+1)	{1, 2, 3, 5}	{7, 9, 11, 13}	{17, 19, 21, 23}	√	0.09	2.47	71.90

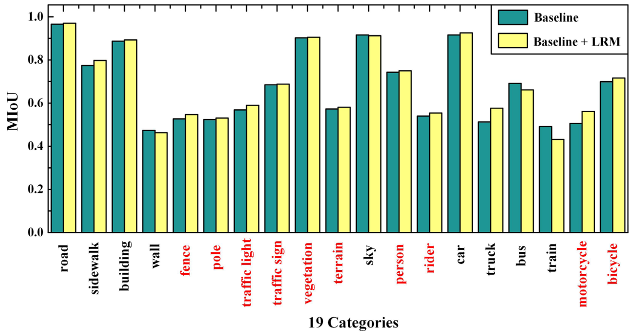


Fig. 9. The performance of Baseline and Baseline+LRM with respect to 19 categories on Cityscapes validation set. Note that the categories in Red refer to relatively small objects.

TABLE IX

ABLATION STUDIES ON MCIMS. MCIM(C): DSP BLOCKS IN CASCADE. MCIM(P): DSP BLOCKS IN PARALLEL

Model	MCIM(C)		MCIM(P)	MIoU(%)
	DSP(C)	GAP		
Baseline				67.84
	√			71.90 (4.06▲)
			√	72.42 (4.58▲)
	√	√		72.14 (4.30▲)

condition. Based on the results in Table VIII, we draw several conclusions. On the one hand, with the number of path n increases, network parameters and computational complexity gradually decrease and eventually suffer from the accuracy degradation from 70.89% MIoU to 70.23% MIoU. These results illustrate that excessive channel reduction could damage the accuracy performance due to large amounts of information loss. On the other hand, adopting diverse sampling rates to capture multi-scale local context information near the target in a dense manner yields 0.69% MIoU boost, which proves that the proposed dense feature sampling strategy delivers a stronger capacity of feature representations in DSP block. Furthermore, the proposed DSP block where both local and global context information are jointly encoded yields about 0.32% MIoU improvement compared with the one without image-level features. According to these observations, the optimal setting for n is 4 and r is 0.2, and the proposed DSP block achieves high accuracy of 71.90% MIoU with only 0.09 M parameters and 2.47 GFLOPs increase.

TABLE X

PERFORMANCE COMPARISONS WITH OTHER STATE-OF-THE-ART MULTI-SCALE MODULES. FPT: FORWARD PASS TIME. ♯PARAMS: THE NUMBER OF PARAMETERS. NOTE THAT FPT AND FLOPS ARE ESTIMATED ON A 320 × 256 × 128 INPUT

Model	FPT(ms)	♯Params(M)	FLOPs(G)	MIoU(%)
Vortex [31]	111.33	5.54	230.29	72.24
ASPP [9]	67.92	4.52	174.86	70.87
FPA [10]	28.49	7.48	43.30	71.92
PPM [7]	19.28	1.28	25.21	71.00
MCIM(P)	27.79	0.48	15.06	72.42
MCIM(C)	28.58	0.12	2.48	72.14

TABLE XI

ABLATION STUDIES ON LRM AND MCIM(C). LRM: LONG-SKIP REFINEMENT MODULE. MCIM(C): MULTI-SCALE CONTEXT INTEGRATION MODULE

Baseline	LRM	MCIM(C)	MIoU(%)
√			67.84
√	√		68.68 (0.84▲)
√		√	72.14 (4.30▲)
√	√	√	72.95 (5.11▲)

3) Ablation for Multi-Scale Context Integration Module: As shown in Table IX, the MCIM(C) denotes that the DSP block are designed in cascade with global information and the MCIM(P) denotes that the DSP blocks are designed in parallel without global information. The model based on MCIM(P) outperforms the model based on DSP(C) module by 0.52% MIoU under the same training conditions, which indicates that the cascaded structure suffers from the feature degradation as the network goes deeper. To overcome this problem, we incorporate image-level features into the DSP(C) module to enhance the model learning capacity. It is clear that the model based on MCIM(C) achieves 72.14% MIoU eventually. In the following, the overall performance of both MCIM(C) and MCIM(P) will be discussed.

4) Comparison With Similar Multi-Scale Modules: In this subsection, we investigate several powerful yet computationally expensive multi-scale modules and compare them with the proposed MCIMs. The last two rows in Table X reveals that MCIMs drastically reduce the number of parameters and simplify the computational complexity while maintaining the competitive accuracy. Specifically, MCIMs achieve speed-faster performance with about 46 times decrease in parameters and about 93 times reduction in FLOPs compared with

TABLE XII
PERFORMANCE COMPARISONS ON CITYSCAPES TEST SET

Method	Backbone	Resolution	ES(FPS)	FLOPs(G)	♯Params(M)	MIoU(%)
SegNet [4]	/	640 × 360	14.6	286.0	29.5	56.1
ENet [12]	/	640 × 360	135.4	3.8	0.4	58.3
Skip-ShuffleNet† [18]	ShuffleNet V1	1024 × 512	-	6.2	1.0	58.3
SQNet†[11]	/	640 × 360	11.6	-	-	59.8
Skip-MobileNet† [18]	MobileNet V1	1024 × 512	45.0	15.4	3.4	62.4
ERFNet† [16]	/	1024 × 512	41.7	53.5	2.1	68.0
ICNet [17]	/	1024 × 512	46.3	-	26.5	69.5
GUNet† [14]	/	1024 × 512	33.3	-	-	70.4
LW-RefineNet101† [15]	ResNet101	512 × 512	55.0	52.0	46.0	72.1
LinkNet† [6]	ResNet18	640 × 360	65.8	21.2	11.5	72.6*
L-DenseNet121† [42]	ResNet18	1024 × 448	31.0	9.8	-	72.8*
DSPNet† [43]	ResNet50	1024 × 512	14.0	-	144.4	64.9*
FCN8s†[22]	VGGNet16	1024 × 512	2.0	136.2	134.5	65.3
Dilation10†[44]	VGGNet16	1024 × 512	0.3	-	140.8	67.1
DRN-C-26 [45]	ResNet50	1024 × 512	-	355.2	20.6	68.0*
LRR-4x [46]	VGGNet16	1024 × 512	-	-	-	69.7
DeepLab V2 [8]	ResNet101	1024 × 512	0.3	457.8	44.0	70.4
DLC [47]	IRNet	1024 × 512	-	26.5	-	71.1
FRRNet [48]	/	1024 × 512	0.3	235.0	-	71.8
RefineNet [5]	ResNet101	1024 × 512	0.9	118.1	-	73.6
DenseASPP121 [29]	DenseNet121	1024 × 512	-	155.8	28.6	76.2
PSPNet [7]	ResNet101	713 × 713	0.8	412.2	250.8	78.4
CIFReNet†(Ours)	MobileNet V2	640 × 360	62.5	7.3	1.9	72.9*/70.9
		512 × 512	55.6	8.3		
		713 × 713	33.3	16.3		
		1024 × 448	38.5	14.4		
		1024 × 512	34.5	16.5		

Vortex module and ASPP module. In the case of FPA module and PPM module, both of them deliver similar accuracy to MCIMs, while suffering from heavier overheads (at least 11 times parameters and 10 times computational complexity increase). These results indicate that current multi-scale modules face the challenge of large amounts of time-consuming operations and invalid computation. Furthermore, it is obvious that the model based on MCIM(P) obtains a bit better accuracy than the model based on MCIM(C), because the former establishes a wider multi-scale module which is helpful to improve the network learning capacity (as demonstrated in [41]). Nevertheless, the "deep and thin" design of MCIM(C) achieves 4 times parameters decrease and 6 times FLOPs reduction compared with MCIM(P). These comparisons illustrate that the MCIM(C) can more efficiently encode multiple context information to generate accurate results, and MCIM(P) is regarded as a sub-optimal choice. Fig. 10 presents some visual examples where some confusion categories exist. Obviously, some analogous objects (e.g., walls and buildings, trucks and buses) are correctly identified by the proposed MCIM.

Finally, we present the comparison of the baseline network and CIFReNet in terms of loss and accuracy curves on CityScapes validation set, as illustrated in Fig. 11(a) and Fig. 11(b). It reveals that CIFReNet could achieve a better convergence and higher accuracy than the baseline network. As shown in Table XI, CIFReNet outperforms the baseline network by achieving a total of 5.11% MIoU promotion. In summary, both qualitative and quantitative results verify that the proposed two modules can effectively enhance the network learning capacity.

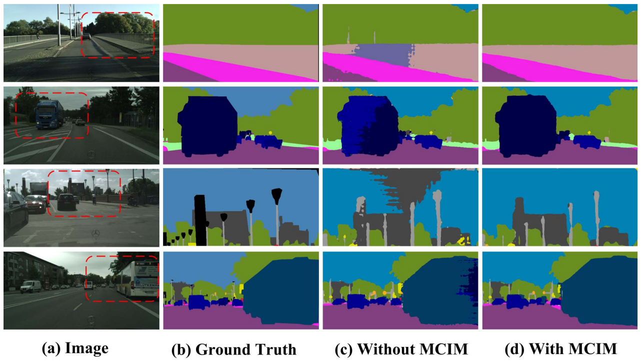
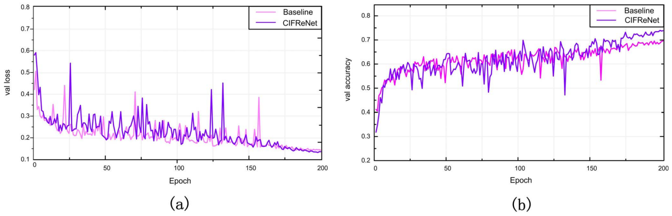

(a) Image **(b) Ground Truth** **(c) Without MCIM** **(d) With MCIM**

Fig. 10. Visualization results of MCIM on Cityscapes validation set. From left to right: Image, Ground Truth, Without MCIM, With MCIM.



(a) (b)

Fig. 11. The validation loss curve and accuracy curve of CIFReNet during 200 training epochs on CityScapes validation set. (a) Validation loss vs. Epoch. (b) Validation accuracy vs. Epoch.

D. Evaluation on Cityscapes Dataset

Besides the above ablation studies, we also compare CIFReNet with other state-of-the-art methods on Cityscapes test set, as displayed in Table XII. "$*$" represents the results

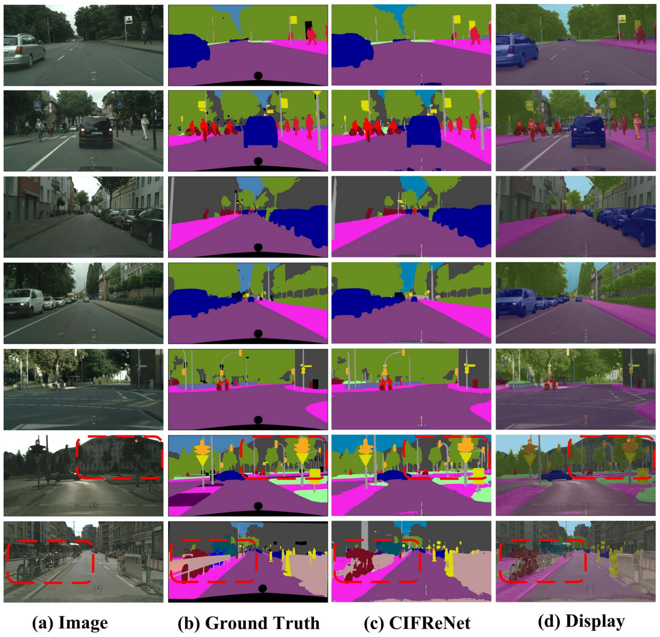

(a) Image (b) Ground Truth (c) CIFReNet (d) Display

Fig. 12. Qualitative results on Cityscapes validation dataset when employing our best model. The last two rows show some failure cases.

evaluated on Cityscapes validation set. Execution Speed (ES) is measured on NVIDIA TITAN X GPU cards. Methods marked with "†" represents the ES measured on other GPU cards. "-" indicates that prior works have not published the corresponding value.

1) Performance Comparisons With Efficiency-Oriented Methods: Previous lightweight methods such as SegNet [4], LinkNet [6], and SQ [11] have accelerated with great pace in speed, while failing to provide an accurate scene description for object parsing. Besides, ENet [12] and Skip-MobileNet [18] achieve excellent performance on model efficiency (e.g., execution speed, memory footprint or computational complexity). However, these methods only pursue efficiency by heavily compressing model, sacrificing too much segmentation accuracy. In contrast, the proposed CIFReNet, which obtains 70.9% MIoU and yields a real-time speed of 62.5 FPS on a 640 × 360 resolution image, makes an obvious performance improvement on both accuracy and efficiency.

Recent lightweight methods (e.g., GUNet [14], LW-RefineNet [15], ERFNet [16], ICNet [17], and L-DenseNet121 [42]) have made a good trade-off between accuracy and speed, but deploying these models on edge devices is difficult due to heavy memory footprint or computational complexity. Differently, CIFReNet takes full advantages of compression techniques to relieve the resource burden, which is more lightweight and resource-saving. Particularly, the number of parameters and computational complexity are significantly reduced to 1.9 M and 7.3 GFLOPs. Besides, CIFReNet outperforms almost all methods stated above in terms of accuracy.

2) Performance Comparisons With Accuracy-Oriented Methods: Table XII also displays the results of CIFReNet compared with high-accuracy methods. We can observe that PSPNet [7], RefineNet [5], and Dilation10 [44] employ costly VGGNets-like or ResNets-like base networks, which take more than 2 seconds for model inference. In contrast, CIFReNet achieves a significant progress in both speed and computation cost when dealing with a high resolution image, and obtains similar or slightly better accuracy compared with most of them (e.g., DeepLab V2 [8], DLC [47], and FRRNet [48]). Without any post-processing operations (e.g., Dense CRF [8]), the proposed CIFReNet reaches a better trade-off among overall performance. Fig. 12 presents some visual examples on Cityscapes validation set. As seen, CIFReNet suffers from commonly challenging issues such as smooth boundary or obscure objects, which we would deal with in future works.

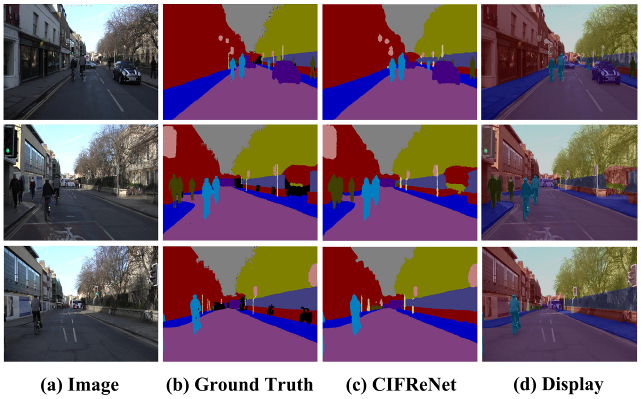

(a) Image (b) Ground Truth (c) CIFReNet (d) Display

Fig. 13. Qualitative results on CamVid validation dataset when employing our best model.

TABLE XIII
PERFORMANCE COMPARISONS ON CAMVID TEST SET

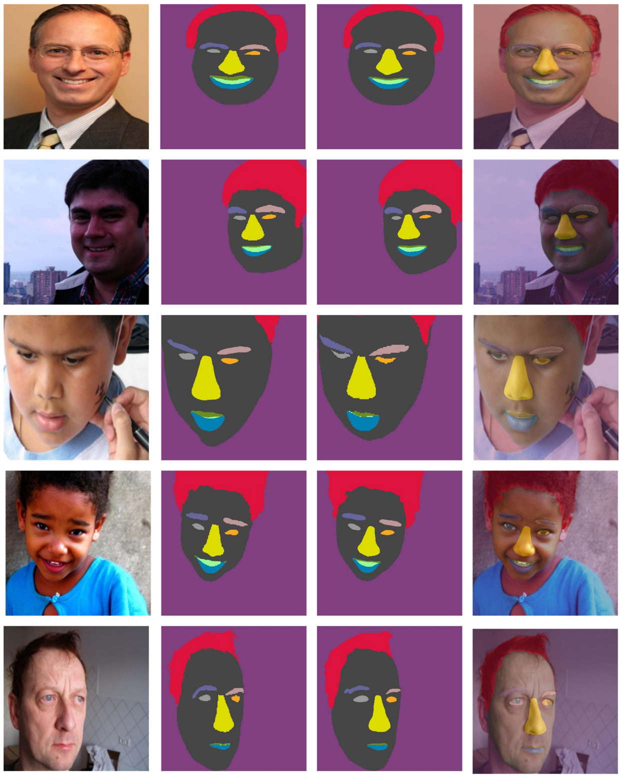
Module	♯Params(M)	MIoU(%)
SegNet [4]	29.5	46.4
DeconvNet [49]	252.0	48.9
ENet [12]	0.4	51.3
LinkNet [6]	11.5	55.8
FCN8s [22]	134.5	57.0
Skip-MobileNet [18]	3.4	58.8
FC-DenseNet56 [50]	1.5	58.9
DeepLab-LFOV [51]	37.3	61.6
Dilation10 [44]	140.8	65.3
CIFReNet(Ours)	1.9	64.5

E. Evaluation on CamVid Dataset

We conduct another experiment on CamVid test set to further evaluate the performance of CIFReNet. The image resolution in CamVid dataset is much smaller than the one in Cityscapes dataset, so we set the dilation rates D_s, D_m, and D_l of three DSP blocks to {1, 2, 3, 5}, {5, 7, 9, 11}, and {11, 13, 15, 17}, in order to effectively gather information from low-resolution feature maps. As reported in Table XIII, CIFReNet achieves 64.5% MIoU with only 1.9 M parameters, which reaches a better trade-off between accuracy and efficiency than accuracy-oriented (e.g., Dilation10 [44], DeepLab-LFOV [51]) methods or efficiency-oriented

TABLE XIV
PERFORMANCE COMPARISONS ON HELEN TEST SET

Module	♯Params(M)	MIoU(%)
FCN8s [22]	134.5	40.7
ENet [12]	0.4	48.2
SegNet [4]	29.5	56.7
ESPNet [13]	0.2	60.7
LinkNet [6]	11.5	63.0
CGNet [52]	0.5	66.3
DeepLab-LFOV [51]	37.3	69.4
ERFNet [16]	2.1	70.1
DeepLab V2 [8]	44.0	70.9
CIFReNet(Ours)	1.9	71.3



(a) Image **(b) Ground Truth** **(c) CIFReNet** **(d) Display**

Fig. 14. Qualitative results on Helen test dataset when employing our best model.

(e.g., SegNet [4], ENet [12], and Skip-MobileNet [18]). Some qualitative results on CamVid validation set are shown in Fig. 13.

F. Evaluation on Helen Dataset

In order to verify the generalization of CIFReNet, we carry out some additional experiments to compare our method with current state-of-the-art methods on Helen test set. We observe that the size of image resolution in Helen dataset is generally about 400 × 400, so the value of D_s, D_m and D_l of three DSP blocks are set to {1, 2, 3, 5}, {3, 5, 7, 11}, and {7, 9, 11, 13} to encode more useful contextual information. Semantic segmentation performance are specifically reported in Table XIV. As seen, CIFReNet achieves 71.3% MIoU with only 1.9 M parameters, which outperforms all methods mentioned above. These results indicate that our method can also make a promising trade-off between accuracy and efficiency for facial object parsing. Fig. 14 presents some prediction visualizations of CIFReNet.

V. CONCLUSION

In this paper, we propose a lightweight Context-Integrated and Feature-Refined Network (CIFReNet) for object parsing. Specifically, our method consists of two core components: Long-skip Refinement Module (LRM) and Multi-scale Contexts Integration Module (MCIM). The LRM is designed to provide a highway for low-frequency information learning and boost the feature refinement efficiently. The MCIM utilizes Dense Semantic Pyramid (DSP) blocks and global features to encode diverse contextual information and obtain larger field of view in an economical way. Experiments show that the proposed CIFReNet not only achieves precise segmentation results but also relieves the burden of model efficiency, which makes it of great potentiality for deployment on resource-constrained intelligent devices.

In future works, we will make further discussions about two aspects: ***a***) Regardless of the dataset size, a scale-adaptive model will be designed to achieve robust performance at a faster speed. ***b***) The issue on how to effectively enhance the performance of boundary prediction will be further explored, which is supposed to replace previous costly measures.

REFERENCES

[1] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, “A review on deep learning techniques applied to semantic segmentation,” in *Proc. 27th Int. Conf. Comput. Linguistics (COLING)*, Aug. 2018, pp. 2132–2144.

[2] W. Hong, Z. Wang, M. Yang, and J. Yuan, “Implicit domain adaptation with conditional generative adversarial networks for depth prediction in endoscopy,” in *Proc. 31th IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1335–1344.

[3] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in *Proc. The 18th Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, Oct. 2015, pp. 234–241.

[4] V. Badrinarayanan, A. Handa, and R. Cipolla, “SegNet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise Labelling,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[5] G. Lin, A. Milan, C. Shen, and I. Reid, “RefineNet: Multi-path refinement networks for high-resolution semantic segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5168–5177.

[6] A. Chaurasia and E. Culurciello, “LinkNet: Exploiting encoder representations for efficient semantic segmentation,” in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2017, pp. 1–4.

[7] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6230–6239.

[8] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[9] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, “Encoder-decoder with Atrous separable convolution for semantic image segmentation,” in *Proc. 15th Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 833–851, 2018.

[10] H. Li, P. Xiong, J. An, and L. Wang, “Pyramid attention network for semantic segmentation,” in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, Sep. 2018, p. 285.

[11] M. Treml, J. Arjona-Medina, T. Unterthiner, R. Durgesh, F. Friedmann, and P. Schuberth, “Speeding up semantic segmentation for autonomous driving,” in *Proc. 30th Conf. Neural Inf. Process. Syst. Workshop (NIPSW)*, Dec. 2016, p. 7.

[12] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," 2016, *arXiv:1606.02147*. [Online]. Available: http://arxiv.org/abs/1606.02147

[13] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, and H. Hajishirzi, "ESPNet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in *Proc. 15th Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 561–580.

[14] D. Mazzini, "Guided Upsampling network for real-time semantic segmentation," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, Sep. 2018, p. 117.

[15] V. Nekrasov, C. Shen, and I. Reid, "Light-weight RefineNet for real-time semantic segmentation," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, Sep. 2018, p. 125.

[16] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "ERFNet: Efficient residual factorized ConvNet for real-time semantic segmentation," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 263–272, Jan. 2018.

[17] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Proc. 15th Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 418–434.

[18] M. Siam, M. Gamal, M. Abdel-Razek, S. Yogamani, and M. Jagersand, "RTSeg: Real-time semantic segmentation comparative study," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 1603–1607.

[19] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[20] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1800–1807.

[21] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.

[22] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, May 2016.

[23] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[24] M. Everingham, "The PASCAL visual object classes challenge (VOC2007) Results," *Lecture Notes Comput. Sci.*, vol. 111, no. 1, pp. 98–136, Jan. 2007.

[25] D. Guo, L. Zhu, Y. Lu, H. Yu, and S. Wang, "Small object sensitive segmentation of urban street scene with spatial adjacency between object classes," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2643–2653, Jun. 2019.

[26] R. Zhang, S. Tang, M. Lin, J. Li, and S. Yan, "Global-residual and local-boundary refinement networks for rectifying scene parsing predictions," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 3427–3433.

[27] H. Ding, X. Jiang, B. Shuai, A. Q. Liu, and G. Wang, "Context contrasted feature and gated multi-scale aggregation for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2393–2402.

[28] R. Zhang, S. Tang, Y. Zhang, J. Li, and S. Yan, "Scale-adaptive convolutions for scene parsing," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2050–2058.

[29] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "DenseASPP for semantic segmentation in street scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3684–3692.

[30] M. Kampffmeyer, N. Dong, X. Liang, Y. Zhang, and E. P. Xing, "ConnNet: A long-range relation-aware pixel-connectivity network for salient segmentation," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2518–2529, May 2019.

[31] C.-W. Xie, H.-Y. Zhou, and J. Wu, "Vortex pooling: Improving context representation in semantic segmentation," 2018, *arXiv:1804.06242*. [Online]. Available: http://arxiv.org/abs/1804.06242

[32] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognit. Lett.*, vol. 30, no. 2, pp. 88–97, Jan. 2009.

[33] V. Le, J. Brandt, Z. Lin, L. D. Bourdev, and T. S. Huang, "Interactive facial feature localization," in *Proc. 12th Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2012, pp. 679–692.

[34] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," in *Proc. 3th Int. Conf. Learn. Represent. (ICLR)*, May 2015.

[35] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proc. 15th Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 294–310.

[36] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. 31th IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7132–7141.

[37] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal Covariate shift," in *Proc. 32th Int. Conf. Mach. Learn. (ICML)*, Jul. 2015, pp. 448–456.

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.

[39] P. Wang *et al.*, "Understanding convolution for semantic segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1451–1460.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[41] S. Zagoruyko and N. Komodakis, "Wide residual networks," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, Sep. 2016, p. 87.

[42] J. Krapac and I. K. S. Segvic, "Ladder-style DenseNets for semantic segmentation of large natural images," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 238–245.

[43] L. Chen, Z. Yang, J. Ma, and Z. Luo, "Driving scene perception network: Real-time joint detection, depth estimation and semantic segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1283–1291.

[44] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*. [Online]. Available: http://arxiv.org/abs/1511.07122

[45] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 636–644.

[46] G. Ghiasi and C. C. Fowlkes, "Laplacian pyramid reconstruction and refinement for semantic segmentation," in *Proc. 13th Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 519–534.

[47] X. Li, Z. Liu, P. Luo, C. C. Loy, and X. Tang, "Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6459–6468.

[48] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe, "Full-resolution residual networks for semantic segmentation in street scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3309–3318.

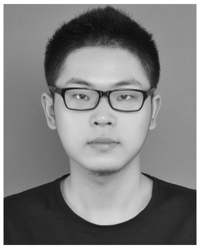
[49] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1520–1528.

[50] S. Jegou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio, "The one hundred layers tiramisu: Fully convolutional DenseNets for semantic segmentation," in *Proc. 30th IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1175–1183.

[51] L. C. Chen, G. G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, May 2015.

[52] T. Wu, S. Tang, R. Zhang, and Y. Zhang, "CGNet: A light-weight context guided network for semantic segmentation," 2018, *arXiv:1811.08201*. [Online]. Available: http://arxiv.org/abs/1811.08201

Bin Jiang received the B.A. degree in mathematics and the M.E. degree in soft engineering from Hunan University, Changsha, China, in 1993 and 2006, respectively, and the Ph.D. degree in computational intelligence and systems science from the Tokyo Institute of Technology, Tokyo, Japan, in 2015. He is currently an Associate Professor with the College of Computer Science and Electronic Engineering, Hunan University. He is a member of ACM and CCF. His research interests include computer vision, machine learning, big data technology, intelligent computing, and deep learning.



Wenxuan Tu received the bachelor's degree in network engineering from Hainan University in 2017. He is currently pursuing the M.E. degree with the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. His research interests include computer vision and machine learning.

Chao Yang received the B.E. and M.E. degrees in computer science from Hunan University, Changsha, China, in 1999 and 2005 respectively, and the Ph.D. degree in computational intelligence and systems science from the Tokyo Institute of Technology, Tokyo, Japan, in 2010. She has ever worked as a Post-Doctoral Fellow with the Tokyo Institute of Technology. Since 2016, she has been an Associate Professor with the College of Computer Science and Electronic Engineering, Hunan University. She is a member of ACM and CCF. Her research interests include computer vision, intelligent computing, machine learning, and deep learning.

Junsong Yuan (Senior Member, IEEE) received the bachelor's degree from the Huazhong University of Science and Technology, Wuhan, China, in 2002, under the Special Class for the Gifted Young Program, the M.Eng. degree from the National University of Singapore, and the Ph.D. degree from Northwestern University. He was an Associate Professor with Nanyang Technological University (NTU), Singapore. He is currently an Associate Professor with the Computer Science and Engineering Department, The State University of New York at Buffalo. His research interests include computer vision, video analytics, gesture and action analysis, large-scale visual search, and mining. He is also the Program Co-Chair of ICME'18 and VCIP'15 and the Area Chair of ACM MM'18, ICPR'18, CVPR'17, ICIP'18'17, and ACCV'18'14. He has served as a Guest Editor for the *International Journal of Computer Vision*. He is also a Senior Area Editor of the *Journal of Visual Communications and Image Representations* and an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEM FOR VIDEO TECHNOLOGY.