electronics

MDPI

Article

LACTNet: A Lightweight Real-Time Semantic Segmentation Network Based on an Aggregated Convolutional Neural Network and Transformer

Xiangyue Zhang, Hexiao Li *, Jingyu Ru, Peng Ji and Chengdong Wu

Faculty of Robot Science and Engineering, Northeastern University, Shenyang 110819, China
* Correspondence: 2101986@stu.neu.edu.cn

Abstract: Transformers have demonstrated a significant advantage over CNNs in modeling long-range dependencies, leading to increasing attention being paid towards their application in semantic segmentation tasks. In the present work, a novel semantic segmentation model, LACTNet, is introduced, which synergistically combines Transformer and CNN architectures for the real-time processing of local and global contextual features. LACTNet is designed with a lightweight Transformer, which integrates a specially designed gated convolutional feedforward network, to establish feature dependencies across distant regions. A Lightweight Average Feature Bottleneck (LAFB) module is designed to effectively capture spatial detail information within the features, thereby enhancing segmentation accuracy. To address the issue of spatial feature loss in the decoder, a long skip-connection approach is employed through the designed Feature Fusion Enhancement Module (FFEM), which enhances the integrity of spatial features and the feature interaction capability in the decoder. LACTNet is evaluated on two datasets, achieving a segmentation accuracy of 74.8% mIoU and a frame rate of 90 FPS on the Cityscapes dataset, and a segmentation accuracy of 71.8% mIoU with a frame rate of 126 FPS on the CamVid dataset.

Keywords: real-time semantic segmentation; convolutional neural network; transformer

check for updates

Citation: Zhang, X.; Li, H.; Ru, J.; Ji, P.; Wu, C. LACTNet: A Lightweight Real-Time Semantic Segmentation Network Based on an Aggregated Convolutional Neural Network and Transformer. *Electronics* **2024**, *13*, 2406. https://doi.org/10.3390/electronics13122406

Academic Editor: Fabio Grandi

Received: 8 May 2024
Revised: 6 June 2024
Accepted: 18 June 2024
Published: 19 June 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The requirement for real-time semantic segmentation tasks is that the model can achieve high-precision image segmentation at speeds exceeding real-time (30FPS), with the core challenge being the model's ability to efficiently capture comprehensive and rich spatial and semantic feature information. By realizing image perception from concrete to abstract, the real-time pixel-level segmentation of the input image is achieved. Indeed, this attribute enables models to distinctly identify and predict the nature of various entities within an image, pinpointing their exact locations and meticulously delineating their outlines. Such refined perceptual accuracy significantly augments the model's proficiency in comprehending and analyzing the intricacies of scenes. As deep learning technology advances unceasingly, semantic segmentation techniques have rapidly evolved. Semantic segmentation algorithms, particularly those centered around Convolutional Neural Networks (CNNs) [1–3], have achieved notable improvements in segmentation precision, allowing for their extensive application in the real world.

However, these models possess a limited receptive field, which poses a disadvantage for learning the long-range dependencies associated with the same object in an image. To address this issue, some models have begun to experiment with increasing the size of convolutional kernels [4], employing techniques such as dilated convolutions [5] and pyramid structures [6] to expand the receptive field dimensions. For instance, LRDNet [7] and ICNet [8] innovatively integrated a multi-scale pyramid design to increase the receptive field, thereby enabling the extraction of feature information at multiple resolutions. LEDNet [9] utilized channel splitting and shuffling and designed an attention-based pyramid feature

extraction approach to reduce computational cost. This method significantly enriches the contextual understanding required for fine-grained segmentation tasks. ERPNet [10] constructed a feature branch that retained high-resolution information through dilated convolutions to capture spatial features. NDNet [11] enhanced the efficiency of semantic segmentation by designing a narrow and deep backbone network. MSCFNet [12] devised an asymmetric encoder–decoder architecture and fused multi-scale context information to increase the receptive field size while reducing the number of parameters. MLFNet [13] constructed a lightweight backbone with a larger receptive field and a better multi-scale context representation performance to encode pixel-level features and designed a spatial compensation branch to gradually reduce the dimensionality of feature maps while preserving target boundaries and contours. DFANet [14] began with a single lightweight backbone and, based on multi-scale feature fusion, increased the receptive field size. Although these methods expand the receptive field and ensure inference speed, they might lead to the loss of long-range connections between features. The loss-associated features further affect the final segmentation accuracy. Therefore, FDDWNet [15] employed skip connections for feature exchange among multiple branches, collecting contextual information from different convolutional layers. Bisenet-V2 [16] obtained different feature information by designing a detail branch and a semantic branch and used a guidance aggregation layer to enhance interconnections and fuse these two types of feature representations. Some researchers have attempted to integrate the Transformer architecture into semantic segmentation tasks to enhance global context modeling and improve long-range associations between features. These methods effectively increased the segmentation accuracy but also added to the computational complexity of the model, reducing its operational speed.

In this work, a novel semantic segmentation network is introduced, which integrates the strengths of Convolutional Neural Networks and Transformer mechanisms into a real-time model, designated as LACTNet. As illustrated in Figure 1, LACTNet utilizes CNNs to acquire rich spatial feature information, while Transformers are employed to capture global context features, reducing the limitations of long-distance associations between features. We employ a feature separation and recombination strategy to reduce the parameter count produced by the Transformer, thereby decreasing computational load and enhancing segmentation efficiency. A gated convolutional feed-forward network is introduced to augment the Transformer's capacity for leveraging local spatial features, thereby boosting the segmentation accuracy of the model. Additionally, we have formulated a Feature Fusion Enhancement Module (FFEM) that employs spatial details from the encoder to enrich and direct the decoder's features, consequently elevating the precision of the model's semantic segmentation.

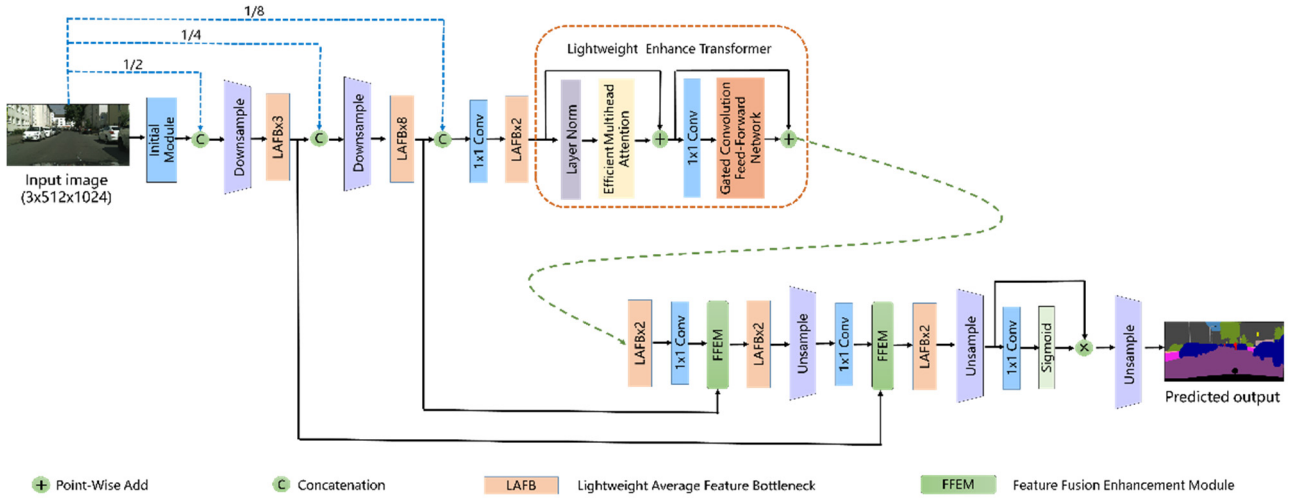


Figure 1. The complete architecture of the proposed LACTNet.

The main contributions can be summarized as follows:

1. A Lightweight Average Feature Bottleneck (LAFB) is designed for the extraction of semantic and spatial features. LAFB utilizes a combination of dilated convolutions and depth-wise separable convolutions, maximizing the richness of the extracted feature set while minimizing the number of parameters and the computational demand of the model.
2. By integrating a gated convolutional feed-forward network (GCFN) and a lightweight Transformer, a Lightweight Enhanced Transformer (LET) is introduced, which improves the model's ability to effectively utilize local spatial features, reduces the number of parameters, and simultaneously enhances the accuracy of segmentation results.
3. A Feature Fusion Enhancement Module (FFEM) is proposed, which excels at leveraging the comprehensive spatial features obtained from the encoder segment and enhances the precision of the model's segmentation capabilities.
4. A real-time semantic segmentation model, LACTNet, is designed by combining a Transformer with a CNN, following an encoder–decoder architecture. Running on a single RTX 4090 platform with only 3.7M parameters, LACTNet achieves an inference speed of 90 FPS and a segmentation accuracy of 74.8% mIoU.

2. Related Work

2.1. Semantic Information and Spatial Information

The key to enhancing the accuracy of model segmentation lies in the integration of high-quality semantic information with spatial information that encompasses numerous image features [17–20]. Many existing models increase segmentation accuracy by integrating semantic information from other stages or by adopting methods that preserve more complete spatial information. For example, JPANet [21] introduced a joint pyramid attention network to enhance the integrity of spatial features, while RELAXNet [22] improved the fusion of different features by facilitating the interaction between local and contextual information. SGCPNet [23] designed a multi-level feature fusion module to capture more contextual information. DSANet [24], targeting the difficulty of capturing spatial features, developed a spatial encoding module to obtain rich spatial information. SFANet [25] considered the association between semantic information and spatial information, designing a Feature Enhancement Block (FEB) to strengthen the correlation between different features. LBN-AA [26] developed a unique Atrous spatial pyramid pool, utilizing pool operations of different sizes to encode rich and unique semantic information to obtain feature information at multiple scales. Although these methods have achieved good results, they do not adequately address the issue of long-range feature connections and the granularity contained in local feature information being lost due to feature extraction operations.

2.2. Attentional Mechanism

Attention mechanisms can selectively capture feature information and have received widespread attention in semantic segmentation. The Convolutional Block Attention Module (CBAM) [27] effectively enhances the utilization of feature information by integrating channel and spatial attention mechanisms. This fused attention system can effectively refine the captured features, emphasizing informative components and eliminating redundancy, leading to a more detailed and efficient representation of feature information. WFDCNet [28] adopts the dual integration of channel and spatial attention mechanisms to enhance the global correlation between features. ATONet [29] integrates channel and spatial attention mechanisms to increase global dependency on information. SAUNet [30] and DANet [31] design an attention module that adaptively enhances spatial information. DARGS [32] proposes an attention residual unit that adaptively captures the characteristics of different channels. MGSeg [33] designs a mixed attention–feature aggregation module to filter out noise spatial details in features, obtain scale-invariant representations, and alleviate the gradient vanishing problem

in early feature learning. In enhancing distant feature associations and global context modeling, the successful application of Transformers has also led to their widespread use [34]. Seaformer [35] designs a compressed Transformer to ensure accuracy while maintaining speed. LETNet [36] designs a U-shaped model architecture that combines CNNs with Transformers to complement each other. However, simply combining these two feature extraction methods may lead to an overemphasis on some information, resulting in the loss of other features.

2.3. Real-Time Semantic Segmentation

Considering the speed requirements for certain application scenarios, an increasing number of real-time semantic segmentation models have emerged [37–40]. BiAttnNet [41] conceptualized a detailed branch that extracted semantic features from target feature mappings. FBSNet [42] employed a dual-branch fusion architecture, which helps to establish and maintain local dependencies between features, thereby enabling the network to capture a wider range of contextual and complex feature information. RoboSeg [43] designed a spatial feature reconstruction branch to replenish the spatial information lost in the feature extraction branch. However, it increased the computational load and consequently reduced the inference speed. DMANet [44] integrated a feature fusion module into the overall structure to preserve the integrity of inactive spatial features. ViT [45] introduced a position-embedding method to guide the segmentation task of input 2D image patches. SETR [46] adopted a sequence-to-sequence perspective, utilizing an entirely Transformer-based structure.

Although Transformer-based models have achieved outstanding segmentation precision, their computational burden is also proportionate to the image size, severely affecting inference speed. Additionally, Transformers in the 2D image domain tend to overemphasize remote pixel dependencies, often resulting in a loss of local feature connections. To overcome the prevalent challenge, we designed a lightweight network that combines Transformers and CNNs to capture local features and better compensate for refined spatial feature details.

3. Proposed Method

3.1. Network Architecture

As illustrated in Figure 1, the proposed LACTNet primarily adopts an encoder–decoder architecture based on Convolutional Neural Networks (CNNs), complemented by a lightweight enhanced Transformer. The CNN extracts a wealth of local feature information to achieve an improved image segmentation performance. The Transformer, integrated with a gated convolutional feed-forward network, captures global contextual features more effectively, enhancing the model's global feature representation. LACTNet employs a skip connection concept similar to that used in Unet [47], which facilitates the exchange between spatial and semantic features, enriching the features of the decoder and aiding in the generation of high-quality segmentation results. The skip connection is represented as Concatenate in Table 1. Furthermore, to achieve a faster training performance and a reduced computational burden, we crop the dataset using a quarter of the original image size as the input image, i.e., an image resolution of $3 \times 512 \times 1024$. The first 3×3 convolutional layer in Table 1 has a stride of 2, thus reducing the input image resolution to 256×512. In Table 1, C represents the number of classes required for segmenting the image target.

Table 1. Detailed architectural configuration of the proposed LACTNet.

	Operator	Mode	Output Size
1	3 × 3 Conv	Stride 2	32 × 256 × 512
2	3 × 3 Conv	Stride 1	32 × 256 × 512
3	3 × 3 Conv	Stride 1	32 × 256 × 512
	Concatenate	-	35 × 256 × 512
4	Downsample	-	64 × 128 × 256
5–7	3 × LAFB	Dilated 2	64 × 128 × 256
	Concatenate	-	67 × 128 × 256
8	Downsample	-	128 × 64 × 128
9–10	2 × LAFB	Dilated 3	128 × 64 × 128
11–12	2 × LAFB	Dilated 7	128 × 64 × 128
13–14	2 × LAFB	Dilated 13	128 × 64 × 128
15–16	2 × LAFB	Dilated 23	128 × 64 × 128
	Concatenate	-	131 × 64 × 128
17	1 × 1 Conv	Stride 1	128 × 64 × 128
18–19	2 × LAFB	Dilated 2	128 × 64 × 128
20	LET		128 × 64 × 128
21	2 × LAFB	Dilated 2	128 × 64 × 128
22	1 × 1 Conv	Stride 1	128 × 64 × 128
23	FFEM	-	128 × 64 × 128
24	2 × LAFB	Dilated 2	128 × 64 × 128
25	UpSample	-	128 × 128 × 256
26	1 × 1 Conv	Stride 1	64 × 128 × 256
27	FFEM	-	64 × 128 × 256
28	2 × LAFB	Dilated 2	64 × 128 × 256
29	UpSample	-	64 × 256 × 512
30	1 × 1 Conv	Stride 1	Cx256 × 512
31	UpSample	-	Cx512 × 1024

3.2. Lightweight Average Feature Bottleneck (LAFB)

As depicted in Figure 2, the designed Local Attention Feature Block (LAFB) leverages a residual module [48] to capture richer feature information within the overall structure. Within LAFB, input features are initially processed by a 3 × 3 convolution that halves the number of channels, effectively reducing the parameter burden. Subsequently, the features are split into two main branches for feature extraction and one branch for feature fusion. The main feature extraction branches are designed to expand the receptive field and minimize the number of parameters, while the guidance branch is meticulously crafted to enhance feature richness.

Due to the nature of depth-wise separable convolutions [49,50], which decompose standard convolutions into two independent steps, depth-wise convolutions and point-wise convolutions, compared to standard convolutions, do not require the computation of all weights in the convolutional kernel but only the weights corresponding to the corresponding channels; they can significantly reduce computational complexity. Considering reducing the computational complexity for each pixel in the input image, we employ depth-wise separable nx1 and 1xn convolutions in the main branches instead of standard convolutions, further alleviating the computational burden of the model and improving the inference speed.

To further enlarge the receptive field of the feature extraction unit, we adopt dilated convolutions [51], which expand the receptive field of the convolutional kernel by adding zero padding around it, enabling it to capture information from further distances. In LAFB, one branch is dedicated to extracting local and short-range feature information, while the other branch uses dilated convolutions with varying dilation rates to achieve a larger receptive field and capture complex long-range feature information.

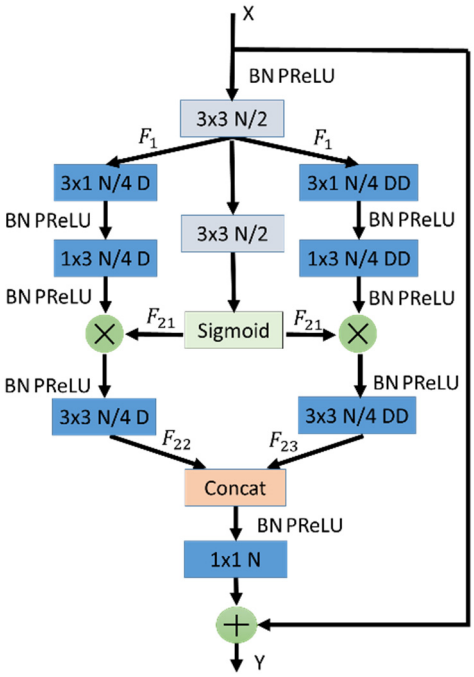


Figure 2. The proposed Lightweight Averaged Feature Block (LAFB) includes depth-wise separable convolutions (D) and depth-wise separable dilated convolutions (DD). N represents the number of channels and, in the figure, we use N/2, N/4, and N to denote the relationships of channel number changes.

The feature-guiding branch within the proposed architecture employs 3×3 standard convolutions in an attempt to recover and refine spatial information that might be lost in the two principal feature extraction branches. This meticulous approach aids in supplementing the detailed textures and edges that are crucial for accurately delineating the objects within an image. Finally, two 3×3 standard convolutions are used for further feature learning and expanding the receptive field. After merging the features captured by the two branches, standard convolutions are employed to restore the original number of channels. For a clearer visual representation of the proposed LAFB, the detailed operations are as follows:

$$F_1 = f_{3\times3}(x), \tag{1}$$

$$F_{21} = f_{Sigmoid}(f_{3\times3}(F_1)), \tag{2}$$

$$F_{22} = f_{3\times3,\mathrm{D}}(f_{1\times3,\mathrm{D}}(f_{3\times1,\mathrm{D}}(F_1)) \times F_{21}), \tag{3}$$

$$F_{23} = f_{3\times3,\mathrm{DD}}(f_{1\times3,\mathrm{DD}}(f_{3\times1,\mathrm{DD}}(F_1)) \times F_{21}), \tag{4}$$

$$y = f_{1\times1}([F_{22}, F_{23}]) + x. \tag{5}$$

where x refers to the image used for feature extraction, y signifies the inactive or less significant feature information, and $f_{k\times k}(\cdot)$ are the convolution operations.

3.3. Lightweight Enhance Transformer (LET)

As previously mentioned, CNNs excel at extracting local features but lack the capacity for global feature representation. To tackle this issue, we incorporate a Transformer module to capture the extensive interactions and dependencies among features. Nevertheless, traditional Transformers generate a substantial computational load. Thus, taking a cue from ETSR [52], we designed a lightweight enhanced Transformer (LET). Unlike ETSR, which uses multiple LETs, we integrated just one LET module to avoid excessive computational demands. The LET sets a reduction layer following the normalization layer, which halves the number of channels to further decrease the computational load.

Drawing inspiration from the principle of group convolutions, LET disaggregates the matrices Q (Query), K (Key), and V (Value) derived from the feature map projections. Q, K, and V are each segmented into 's' parts, and they are employed across the matching segments Qi, Ki, and Vi. By concatenating the outputs Oi obtained from each group, we arrive at the final output O. Channel numbers are subsequently restored through a mapping layer. This approach enables the network to assign dynamic weights to various parts of the input data, allowing it to concentrate on the pertinent information required for the specific task at hand. Using dot product attention in LET can achieve efficient computation and reduce parameter count while maintaining the ability to capture complex input patterns. The detailed structure is demonstrated in Figure 3a, and the dot-product attention formula is presented as follows:

$$O = softmax\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{6}$$

In the dot-product attention mechanism, Q represents the query matrix, K stands for the key matrix, and V is the value matrix. The embedding dimension is denoted by d, which is the size of the query, key, and value vectors. The output matrix is represented by O.

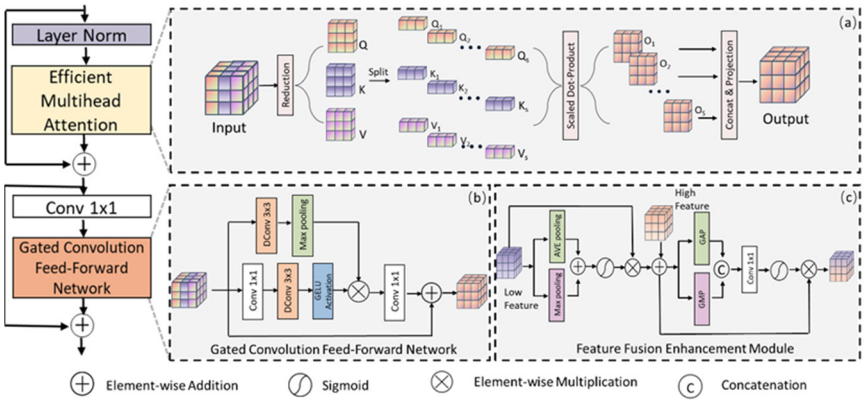


Figure 3. The proposed Lightweight Enhance Transformer (LET) is shown in (**a**,**b**). (**c**) denotes the Feature Fusion Enhancement Module (FFEM).

In traditional Transformers, feed-forward networks typically consist of two 1×1 convolutions interlinked with a non-linear unit. However, this architecture limits the capacity for utilizing local features. As an improvement, we implemented a gated convolutional feed-forward network that utilizes a spatial attention module to model spatial features. The network encodes spatial location information using 3×3 depth-wise separable convolutions alongside max pooling operations, which are adept at capturing the spatial features that need to be emphasized. To seize contextual features, standard 1×1 convolutions and 3×3 depth-wise separable convolutions are employed. As a gating mechanism, the incorporation of the GELU non-linearity allows for the learning of feature interrelations among channels, thus better guiding the spatial feature representation.

The Gated Convolutional Feed-Forward Network (GCFN) was purposefully constructed to improve the extraction of both global and local features. And gating mechanisms provide a way to control information flow and allow the network to focus on the more relevant features of a given task. The spatial attention branch adjusts the network's focus on different spatial regions, strengthens important features, and suppresses less relevant features. The proposed network structure is depicted in Figure 3b, demonstrating how these components are arranged to refine the network's ability to process spatial information.

3.4. Feature Fusion Enhancement Module (FFEM)

In the field of semantic segmentation, features from different stages contain different information. Higher-level features are rich in contextual information due to the aggregation

of cues from large receptive fields, but they often lack complete spatial details. Conversely, lower-level features retain more precise spatial information but are less informative contextually. To effectively leverage the complementary strengths of both features at all levels, many networks employ skip connections for multi-level feature fusion [53–56].

Therefore, we designed a Feature Fusion Enhancement Model (FFEM) to selectively emphasize critical features and guide the fusion process at various stages within the feature maps. The rich semantic features from higher layers can be effectively informed by the intact spatial information present in the lower layers. This selective emphasis and guided fusion help to attain a more harmonious integration of features, resulting in more precise and contextually accurate semantic segmentation.

As shown in Figure 4, the low-level features are introduced into FFEM through long skip connections, enabling the transfer of detailed spatial information directly into the fusion process. By employing both average pooling and max pooling operations, the module effectively captures a wide array of spatial details through average pooled and max pooled features. Global Average Pooling (GAP) compresses the feature map into a vector by performing an average calculation over all the pixels in the feature map. This aids in transforming spatial information into a feature vector that can be used for subsequent fully connected layers, helping to retain the spatial information of the feature map while reducing the number of parameters and the computational complexity. Global Max Pooling (GMP) compresses the feature map into a vector by performing a maximum calculation over all the pixels in the feature map. This operation can preserve the most important feature information, aiding in maintaining the key features while compressing other feature. By combining these two pooling operations, the FFEM can obtain a richer and more effective feature representation. After the pooling operations, 1×1 standard convolutions are applied to obtain channel-wise features. This step is crucial, as it can better fit the nonlinear relationships between the channels. Utilizing channel-wise features through global pooling, followed by 1×1 convolutions, allows the network to improve the relevancy of the features while also reducing computational complexity.

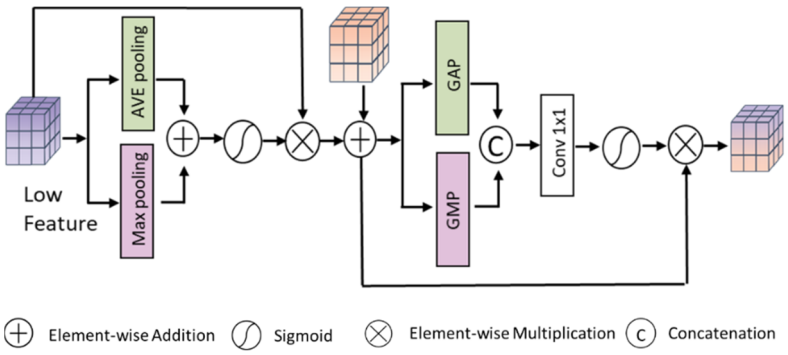


Figure 4. The proposed Feature Fusion Enhancement Module (FFEM).

The dual pooling strategy (average and max) is aimed at different aspects of the input features. The average pooling tends to smooth out the feature response, highlighting the most consistently active features, while the max pooling focuses on the most salient features that stand out. The concurrent use of both strategies, in conjunction with channel-wise processing via 1×1 convolutions, significantly boosts the network's feature representation capabilities while maintaining high efficiency, a critical factor for achieving real-time performance in semantic segmentation tasks. The proposed FFEM can capture both detailed information representing the object edge and the high-level semantic information required of different regions.

4. Experiments

This section provides a comprehensive description of the datasets and the experimental setup. Ablation studies for the proposed LACTNet were performed on the Cityscapes Dataset. Subsequently, the proposed model was benchmarked against state-of-the-art approaches on both the Cityscapes [57] and CamVid [58] datasets, demonstrating the efficacy and advantage of the proposed model.

4.1. Experimental Settings

4.1.1. Cityscapes

The Cityscapes dataset consists of color images captured from German urban streets, featuring high-resolution images, typically 2048 × 1024 pixels. The diverse content of these images includes urban landscapes, roads, traffic signs, pedestrians, and vehicles across 19 categories. This provides a varied training dataset for models but also increases the difficulty of segmentation. The dataset comprises 2975 images for training, 500 images for validation, and 1525 images for testing.

4.1.2. CamVid

The CamVid dataset comprises color images of urban streets in Cambridge, UK, captured from various angles, with a resolution of 720 × 960 pixels. It contains 11 distinct object categories such as buildings, roads, trees, and vehicles. The dataset is divided into 367 images for training, 101 images for validation, and 233 images for testing. The images in the CamVid dataset are relatively small and encompass a limited number of categories. However, the challenges posed by the dataset include occlusions, changes in illumination, and variations in viewpoint within the images.

4.1.3. Settings

We developed our model within the PyTorch 1.7.0 framework and utilized an RTX 4090 for training purposes. Due to the varying number of categories and image resolutions between the two datasets, we employed different optimizers and parameter settings to achieve optimal training outcomes. The two datasets were resized to 3 × 512 × 1024 and 3 × 360 × 480, respectively. For the Cityscapes dataset, we implemented mini-batch stochastic gradient descent (SGD) [59], with a batch size of 8, momentum of 0.9, and weight decay of 1×10^{-4}. We utilized a "poly" learning rate schedule, as expressed in Equation (7), with an initial learning rate of 4.5×10^{-2} and a power of 0.9.

$$lr = lr_{init} \times \left(1 - \frac{iter}{max_iter}\right)^{power} \tag{7}$$

Moreover, we adopted the Online Hard Example Mining (OHEM) loss function as our loss function. OHEM [60] is a technique for optimizing the loss function of neural networks, which is particularly suitable for addressing imbalanced datasets where certain classes appear at much higher frequencies than others. The advantage of OHEM lies in its ability to enhance the model's recognition capability for classes that are less prevalent in the data, thereby mitigating the impact of class imbalance. Because the network does not adopt a pre-training scheme, we set the number of training epochs to 1000 to achieve optimal training results. This extended training period compensated for the absence of pre-trained model weights, giving the network ample opportunity to learn from the data and converge on effective parameters.

For CamVid, we trained with the Adam optimizer, using a momentum of 0.9, an associated weight decay of 2×10^{-4}, and a batch size of 8. The initial learning rate was set to 1×10^{-4}, and we trained all models for 1000 epochs. Inspired by ENet [61], we implemented a class weighting scheme on CamVid to address class imbalance, which is defined as follows:

$$w_{class} = \frac{1}{ln(c + P_{class})} \tag{8}$$

The hyperparameter C was set to 1.10 in the class weighting scheme.

In terms of data augmentation, we applied random horizontal flipping, mean subtraction, and random scaling to the input images. The range of random scaling encompassed factors, {0.75, 1.0, 1.25, 1.5, 1.75, and 2.0}. Data augmentation techniques significantly increase the size of both the training and validation sets, enhancing the model's generalization capability and performance. However, the size and content of the test set remained unchanged to ensure the accuracy of the evaluation. In this way, data augmentation effectively utilizes the image database, improving the model's performance and robustness.

4.2. Ablation Studies

In this section, the performance of each module in the whole network was verified by sufficient ablation experiments. We employed the Cityscapes datasets for experiment effect analysis. The ablation experiments were conducted on Cityscapes training datasets, and Cityscapes validation datasets were used to evaluate the experimental effect.

4.2.1. Ablation Study for LET

In this section, we validate the advantages of LET in enhancing segmentation precision through ablation studies. As shown in Table 2, LET improves segmentation performance by 2.5% over the base model with only a minimal increase in the number of parameters. Although GCFN leads to an increase of 0.02M in parameter count, it achieves a segmentation accuracy of 74.8%, which is significantly higher than the basic model and the version without GCFN, at 72.3% and 74.5% segmentation accuracy, respectively. Although the standard Transformer achieved a segmentation accuracy of 74.7%, its segmentation speed and parameter count are weaker than those of LET. Four ablation experiments demonstrated that LET and its internal component GCFN play a positive role in image segmentation, effectively enhancing the precision of the segmentation capability.

Table 2. Evaluation results of LET on the Cityscapes dataset. The best parameter is marked in bold.

Module	mIoU (%)	Speed (fps)	Params (M)
Base	72.3	**173**	**1.43**
Base + Transformer	74.7	76	4.01
Base + LET (No GCFN)	74.5	91	3.72
Base + LET	**74.8**	90	3.74

4.2.2. Ablation Study for FFEM

In this section, we validate the advantages of the Feature Fusion Enhancement Module (FFEM) in enhancing segmentation precision through ablation experiments. As indicated in Table 3, FFEM results in a 1.5% increase in segmentation accuracy compared to the base model (only no FFEM), while also achieving a real-time segmentation speed. This is primarily due to the long skip connections that incorporate complete spatial features from the encoder, effectively guiding the rich semantic features in the decoder. The spatial attention mechanism and through-channel attention mechanism within FFEM effectively enhance feature integrity and improve inter-channel relationships. Ablation studies have demonstrated that FFEM plays an active role in image segmentation, effectively improving segmentation precision.

Table 3. Evaluation results of different attention mechanism modules on the Cityscapes validation dataset. The best parameter is marked in bold.

Attention Mechanism	mIoU (%)	Speed (fps)	Params (M)
LCANet-No FFEM	73.3	**101**	**3.69**
LCANet	**74.8**	90	3.74

4.2.3. Ablation Study of LAFB

In this section, we compare the performance of the LAFB with several state-of-the-art network bottleneck units. As shown in Table 4, the bottleneck model of the experimental network has the fewest parameters and the highest inference speed but the lowest segmentation accuracy. This is because the bottleneck model has a shallower depth and fewer branches for feature extraction. However, bottleneck units in other experimental networks can achieve better segmentation accuracy due to their deeper model depth and multi-branch structure. Compared to other bottleneck units, LAFB improves spatial feature integrity and model segmentation accuracy by effectively guiding the feature information captured through the two main feature extraction branches. Although LAFB results in a larger number of parameters, it obtains the highest segmentation precision, surpassing second place, occupied by EAB, by 1.6%. Overall, ablation studies affirm the superior performance of LAFB over other comparative units.

Table 4. Evaluation results of different feature extraction units on the Cityscapes validation dataset. The best parameter is marked in bold.

Bottleneck Unit	mIoU (%)	Speed (fps)	Params (M)
Bottleneck	68.2	**109**	**2.77**
DAB	72.7	93	3.74
EAB	73.2	91	3.76
LAFB	**74.8**	90	3.74

4.3. Comparison with Other Lightweight SOTA Methods

The number of categories and image resolutions in the two datasets are not identical. Meanwhile, real-time semantic segmentation demands not only accuracy but also model inference speed. To address this, we opted to crop the images in the datasets to balance these requirements.

4.3.1. Results on Cityscapes

Table 5 presents the Intersection over Union (IoU) results for each category within the Cityscapes test dataset. LACTNet achieves the best results in 11 out of the 19 categories without the aid of pre-training. Notably, it surpasses the second best model by 5.7% in the fence category, 2.2% in trucks, 6.1% in buses, and 6.1% in walls. Although LACTNet does not attain the best segmentation performance in the eight categories of sidewalk, vegetation, terrain, sky, pedestrian, rider, car, and motorcycle, it still manages to achieve moderately high segmentation accuracy. Overall, it achieves the best mean Intersection over Union (mIoU) in the collective results.

Table 5. Per-class IoU (%) results on the Cityscapes test set. "Avg" represents the average results of all these categories. The best results are shown in bold.

Method	Roa	Sid	Bui	Wal	Fen	Pol	Tli	Tsi	Veg	Ter	Sky	Per	Rid	Car	Tru	Bus	Tra	Mot	Bic	Avg
EFRNet [10]	97.9	82.1	90.7	45.2	50.4	59.0	62.6	68.4	91.9	69.4	94.2	78.5	59.8	93.4	52.3	60.8	53.7	49.9	64.2	69.7
DABNet [18]	97.9	82.0	90.6	45.5	50.1	59.3	63.5	67.7	91.8	70.1	92.8	78.1	57.8	93.7	52.8	63.7	56.0	51.3	66.8	70.1
LEDNet [9]	98.1	79.5	91.6	47.7	49.9	62.8	61.3	72.8	92.6	61.2	**94.9**	76.2	53.7	90.9	64.4	64.0	52.7	44.4	71.6	70.6
FBSNet [42]	98.0	83.2	91.5	50.9	53.5	62.5	67.6	71.5	**92.7**	**70.5**	94.4	82.5	63.8	93.9	50.5	56.0	37.6	56.2	70.1	70.9
FDDWNet [15]	98.0	82.4	91.1	52.5	51.2	59.9	64.4	68.9	92.5	70.3	94.4	80.8	59.8	94.0	56.5	68.9	48.6	55.7	67.7	71.5
MSCFNet [12]	97.7	82.8	91.0	49.0	52.5	61.2	67.1	71.4	92.3	70.2	94.3	**82.7**	62.7	94.1	50.9	66.1	51.9	57.6	70.2	71.9
LETNet [36]	98.2	83.6	91.6	50.9	53.7	61.0	66.7	70.5	92.5	70.5	**94.9**	82.3	61.7	94.4	55.0	72.4	57.0	57.6	69.3	72.8
LBN-AA [26]	98.2	**84.0**	91.6	50.7	49.5	60.9	69.0	73.6	92.6	70.3	94.4	83.0	**65.7**	**94.9**	62.0	70.9	53.3	**62.5**	71.8	73.6
LACTNet	**98.3**	83.9	**92.2**	**58.6**	**59.4**	**64.0**	**69.2**	**77.2**	92.3	65.7	94.3	81.1	60.2	94.5	**66.6**	**78.5**	**57.6**	59.8	**75.7**	**74.8**

In Table 6, we quantitatively compare LACTNet with other lightweight semantic segmentation methods on the Cityscapes test dataset. The results indicate that, when the number of parameters is similar, the compared models fall short of LACTNet in both segmentation accuracy and inference speed. Among the models compared, CGNet [17], FBSNet [42], and FDDWNet [15] have smaller parameter counts; however, their segmentation accuracy is significantly lower than that of LACTNet, highlighting a notable performance disparity. Although Seaformer [35] has a similar number of parameters to LACTNet and higher segmentation accuracy, it relies solely on a Transformer-based feature extraction scheme, resulting in an inference speed of only 22 FPS, which is below the real-time requirement of 30 FPS for semantic segmentation. From the perspective of real-time semantic segmentation tasks, Seaformer's performance is inferior to that of LACTNet. In terms of speed, LACTNet operates at 90 FPS, which is a moderate level that exceeds the real-time semantic segmentation requirement of 30 FPS. In terms of FLOPs, LACTNet's computational cost is 37.3 G, which is less than LBN-AA's 49.5 G, achieving higher segmentation precision. Although DFANet [14] has the lowest computational cost, its segmentation performance is 3.5% lower than that of LACTNet. In terms of mean Intersection over Union (mIoU), Seaformer [35] and SegFormer [34] offer higher segmentation accuracies than LACTNet; however, they depend on transformers with enormous computational demands and their inference speeds are far below the real-time threshold, and they thus fail to meet the requirements of real-time semantic segmentation tasks. This validates the effectiveness of designing models that combine Transformers and CNNs for real-time semantic segmentation tasks. LETNet [36] also uses a feature extraction method that combines transformers with CNNs, but its attention-based feature extraction compresses channel feature information, resulting in a segmentation accuracy that is 2.0% lower than LACTNet. In summary, the adoption of LACTNet, which integrates a CNN and Transformer, effectively balances the model's segmentation accuracy, inference speed, and parameter count, efficiently addressing real-time semantic segmentation tasks while maintaining a superior performance. Figure 5 illustrates the color mappings for different semantic categories in the Cityscapes dataset. Figure 6 provides visual segmentations from MGSeg [33], LETNet [36], BisNet-V2 [16], and LACTNet, intuitively showcasing the advantages of our model.

Table 6. Comparison with state-of-the-art image semantic segmentation methods on Cityscapes test dataset. The best results are shown in bold.

Methods	Backbone	Input Size	FPS	Params (M)	FLOPs (G)	mIoU (%)
CGNet [17]	No	360 × 640	74	**0.50**	6.0	64.8
NDNet [11]	No	1024 × 2048	40	0.52	14.0	65.3
EFRNet [10]	No	512 × 1024	44	2.10	21.0	69.7
DABNet [18]	No	512 × 1024	94	0.76	10.5	70.1
LEDNet [9]	No	512 × 1024	40	0.94	11.5	70.6
SGCPNet [23]	No	1024 × 2048	103	0.61	4.5	70.9
FBSNet [42]	No	512 × 1024	90	0.62	9.7	70.9
DFANet [14]	Xception	1024 × 1024	100	7.80	**3.4**	71.3
FDDWNet [15]	No	512 × 1024	60	0.80	85	71.5
MSCFNet [12]	No	512 × 1024	50	1.15	17.1	71.9
MLFNet [13]	ResNet34	512 × 1024	72	3.99	15.54	72.1
BisNet-V2 [16]	No	512 × 1024	**156**	3.40	21.2	72.6
MGSeg [33]	ShuffleNet	1024 × 2048	101	4.50	16.2	72.7
LETNet [36]	No	512 × 1024	150	0.95	13.6	72.8
LBN-AA [26]	No	448 × 896	51	6.20	49.5	73.6
ATONet [29]	No	1024 × 2048	42	13.30	134.5	74.4
Seaformer [35]	No	1024 × 2048	22	4.00	8.0	75.9
SegFormer [34]	Mit-B0	1024 × 2048	12	84.70	125.5	**76.2**
LACTNet	No	512 × 1024	90	3.74	37.3	74.8

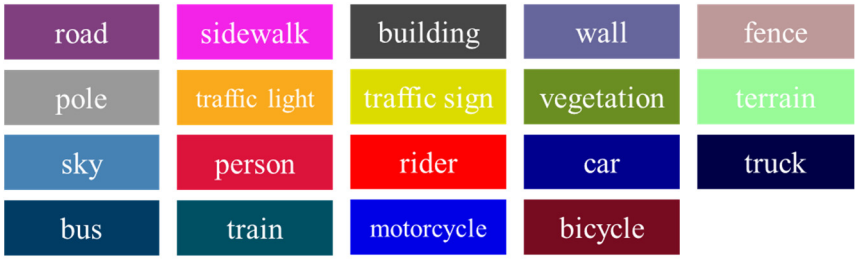


Figure 5. The color distribution of different semantic categories in Cityscapes.

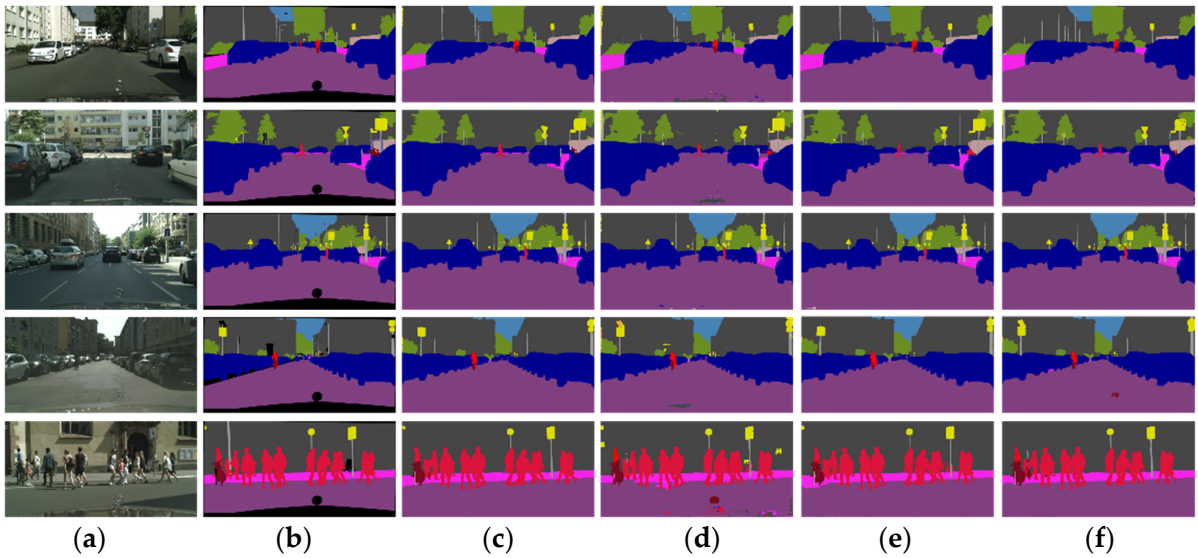


Figure 6. Semantic segmentation results on Cityscapes. (**a**) Original images. (**b**) Ground truth. (**c**) Our LACTNet. (**d**) BisNet-V2. (**e**) LETNet. (**f**) MGSeg.

4.3.2. Results on CamVid

We further evaluated the performance of LACTNet on the CamVid dataset with the outcomes displayed in Table 7. While SegFormer [34] achieves a higher segmentation accuracy than LACTNet, its model has many more parameters than our method. The excessive number of parameters impedes their segmentation efficiency, enabling them to reach only 19 FPS, which falls short of the real-time processing benchmark. In stark contrast, LACTNet achieves a significantly faster inference speed of 126 FPS. Compared to models of a similar size, LACTNet achieves a higher segmentation accuracy. Against larger models, LACTNet accomplishes superior segmentation results with fewer parameters, displaying its superior performance. Figure 7 shows the color mappings for different semantic categories in the CamVid dataset. Figure 8 provides a visual comparison between MGSeg [33], LETNet [36], BisNet-V2 [16], and LACTNet, further emphasizing the advantages of our model. These experiments demonstrate that LACTNet achieves a favorable balance of performance in the field of real-time semantic segmentation.

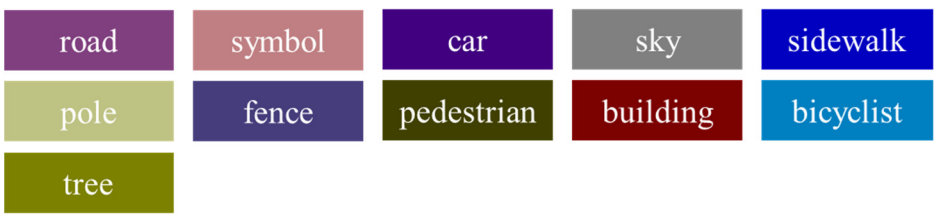


Figure 7. The color distribution of different semantic categories in CamVid.

Table 7. Comparison with state-of-the-art image semantic segmentation methods on CamVid test dataset. The best results are shown in bold.

Methods	Backbone	Input Size	FPS	Params (M)	FLOPs (G)	mIoU (%)
NDNet [11]	No	360 × 480	74	**0.50**	6.0	64.8
DFANet [14]	Xception	720 × 960	40	0.52	14.0	65.3
EFRNet [10]	No	360 × 480	44	2.10	21.0	69.7
CGNet [17]	No	360 × 480	94	0.76	10.5	70.1
DABNet [18]	No	360 × 480	40	0.94	11.5	70.6
LEDNet [9]	No	360 × 480	103	0.61	4.5	70.9
FDDWNet [15]	No	360 × 480	90	0.62	9.7	70.9
LBN-AA [26]	No	720 × 960	100	7.80	**3.4**	71.3
BisNet-V2 [16]	No	720 × 960	60	0.80	85	71.5
FBSNet [42]	No	360 × 480	50	1.15	17.1	71.9
SGCPNet [23]	No	720 × 960	72	3.99	15.54	72.1
MSCFNet [12]	No	360 × 480	**156**	3.40	21.2	72.6
MLFNet [13]	ResNet34	720 × 960	101	4.50	16.2	72.7
ATONet [29]	No	720 × 960	150	0.95	13.6	72.8
LETNet [36]	No	360 × 480	51	6.20	49.5	73.6
MGSeg [33]	ShuffleNet	360 × 480	42	13.30	134.5	74.4
Seaformer [35]	No	720 × 960	22	4.00	8.0	75.9
SegFormer [34]	Mit-B0	720 × 960	12	84.70	125.5	**76.2**
LACTNet	No	360 × 480	90	3.74	37.3	74.8

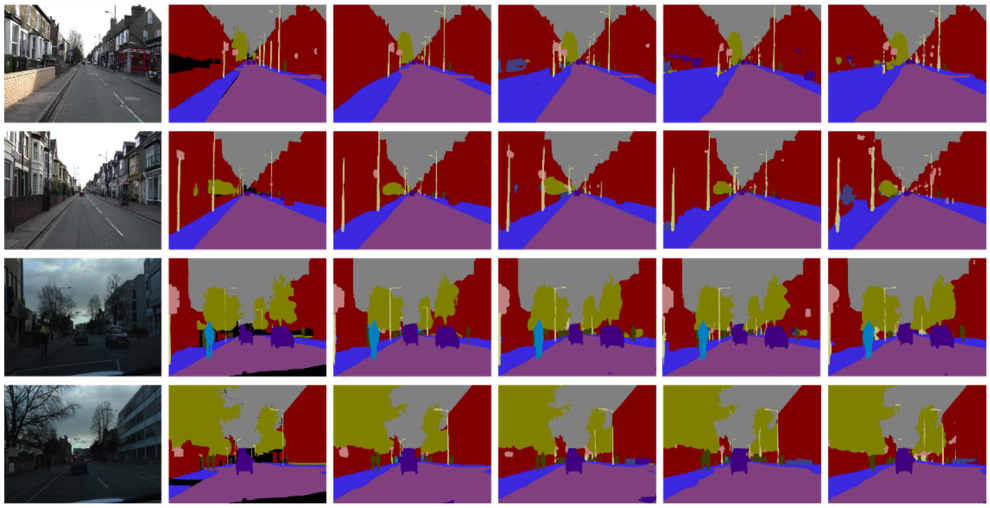


Figure 8. Semantic segmentation results on CamVid. (**a**) Original images. (**b**) Ground truth. (**c**) Our LACTNet. (**d**) BisNet-V2. (**e**) LETNet. (**f**) MGSeg.

5. Results and Discussion

This paper introduces LACTNet, an innovative real-time semantic segmentation model that integrates Transformer and CNN frameworks. LACTNet combines a lightweight Transformer with a gated convolutional feed-forward network as attention units, positioned between the encoder and decoder, to more effectively capture global contextual features. Additionally, a CNN-based LAFB is designed to capture richer feature information while maintaining a lightweight structure. To compensate for the lack of spatial features in the decoder, LACTNet employs long-range connections and designs FFEM to invoke complete spatial features from the encoder, guiding the semantic features in the decoder. The proposed model suppresses noise interference and enhances feature interaction, leading to improved segmentation accuracy. The experimental results demonstrate that the proposed LACTNet achieves an exceptional segmentation performance at real-time inference speeds.

In future research, we will continue to explore how to utilize the characteristics of both CNNs and Transformers to further capture more complete spatial features while ensuring the richness of semantic features, aiming to improve the model's segmentation progress for smaller and more intricate targets. Furthermore, we will investigate the issues of excessive parameter counts and insufficient inference speeds in existing models to develop a real-time semantic segmentation model that better balances parameter count, inference speed, and segmentation accuracy.

Author Contributions: Methodology, H.L.; Software, X.Z. and H.L.; Validation, H.L.; Investigation, X.Z. and J.R.; Resources, P.J.; Writing—original draft, X.Z.; Writing—review & editing, C.W.; Funding acquisition, X.Z., J.R. and P.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China under Grant (62203099 and 62303099); the Natural Science Foundation of Liaoning Province under Grant (2023-MSBA-090 and 2023-MSBA-071); the National Key R&D Program Project of China under Grant (2022YFB4703101); and the Fundamental Research Funds for the Central Universities under Grant (N2326004).

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
2. Poudel, R.P.; Liwicki, S.; Cipolla, R. Fast-scnn: Fast semantic segmentation network. *arXiv* **2019**, arXiv:1902.04502.
3. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef]
4. Peng, C.; Zhang, X.; Yu, G.; Luo, G.; Sun, J. Large kernel matters—Improve semantic segmentation by global convolutional network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4353–4361.
5. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef]
6. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
7. Zhuang, M.; Zhong, X.; Gu, D.; Feng, L.; Zhong, X.; Hu, H. LRDNet: A lightweight and efficient network with refined dual attention decorder for real-time semantic segmentation. *Neurocomputing* **2021**, *459*, 349–360.
8. Zhao, H.; Qi, X.; Shen, X.; Shi, J.; Jia, J. Icnet for real-time semantic segmentation on high-resolution images. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8-14 September 2018; pp. 405–420.
9. Wang, Y.; Zhou, Q.; Liu, J.; Xiong, J.; Gao, G.; Wu, X.; Latecki, L.J. Lednet: A lightweight encoder-decoder network for real-time semantic segmentation. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 1860–1864.
10. Zhao, J.; Zou, F.; Li, R.; Li, Y.; Li, K. Efficient Resolution-preserving Network for Real-time Semantic Segmentation. In Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN), Virtual, 18–22 July 2021; pp. 1–8.
11. Yang, Z.; Yu, H.; Fu, Q.; Sun, W.; Jia, W.; Sun, M.; Mao, Z.-H. NDNet: Narrow while deep network for real-time semantic segmentation. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 5508–5519. [CrossRef]
12. Gao, G.; Xu, G.; Yu, Y.; Xie, J.; Yang, J.; Yue, D. Mscfnet: A lightweight network with multi-scale context fusion for real-time semantic segmentation. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 25489–25499. [CrossRef]
13. Fan, J.; Wang, F.; Chu, H.; Hu, X.; Cheng, Y.; Gao, B. MLFNet: Multi-Level Fusion Network for Real-Time Semantic Segmentation of Autonomous Driving. *IEEE Trans. Intell. Veh.* **2023**, *8*, 756–767. [CrossRef]
14. Li, H.; Xiong, P.; Fan, H.; Sun, J. Dfanet: Deep feature aggregation for real-time semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 9522–9531.
15. Liu, J.; Zhou, Q.; Qiang, Y.; Kang, B.; Wu, X.; Zheng, B. FDDWNet: A lightweight convolutional neural network for real-time semantic segmentation. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 2373–2377.

16. Yu, C.; Gao, C.; Wang, J.; Yu, G.; Shen, C.; Sang, N. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. *Int. J. Comput. Vision* **2021**, *129*, 3051–3068. [CrossRef]
17. Wu, T.; Tang, S.; Zhang, R.; Cao, J.; Zhang, Y. Cgnet: A light-weight context guided network for semantic segmentation. *IEEE Trans. Image Process.* **2020**, *30*, 1169–1179. [CrossRef]
18. Li, G.; Jiang, S.; Yun, I.; Kim, J.; Kim, J. Depth-wise asymmetric bottleneck with point-wise aggregation decoder for real-time semantic segmentation in urban scenes. *IEEE Access* **2020**, *8*, 27495–27506. [CrossRef]
19. Yang, Q.; Chen, T.; Fan, J.; Lu, Y.; Zuo, C.; Chi, Q. Eadnet: Efficient asymmetric dilated network for semantic segmentation. In Proceedings of the ICASSP 2021—2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 2315–2319.
20. Singha, T.; Pham, D.S.; Krishna, A.; Gedeon, T. A lightweight multi-scale feature fusion network for real-time semantic segmentation. In Proceedings of the Neural Information Processing: 28th International Conference, ICONIP 2021, Sanur, Bali, Indonesia, 8–12 December 2021.
21. Hu, X.; Jing, L.; Sehar, U. Joint pyramid attention network for real-time semantic segmentation of urban scenes. *Appl. Intell.* **2022**, *52*, 580–594. [CrossRef]
22. Liu, J.; Xu, X.; Shi, Y.; Deng, C.; Shi, M. RELAXNet: Residual efficient learning and attention expected fusion network for real-time semantic segmentation. *Neurocomputing* **2022**, *474*, 115–127. [CrossRef]
23. Hao, S.; Zhou, Y.; Guo, Y.; Hong, R.; Cheng, J.; Wang, M. Real-time semantic segmentation via spatial-detail guided context propagation. *IEEE Trans. Neural Netw. Learn. Syst.* 2022; *Early Access*.
24. Elhassan, M.A.; Huang, C.; Yang, C.; Munea, T.L. DSANet: Dilated spatial attention for real-time semantic segmentation in urban street scenes. *Expert Syst. Appl.* **2021**, *183*, 115090. [CrossRef]
25. Weng, X.; Yan, Y.; Chen, S.; Xue, J.H.; Wang, H. Stage-aware feature alignment network for real-time semantic segmentation of street scenes. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 4444–4459. [CrossRef]
26. Dong, G.; Yan, Y.; Shen, C.; Wang, H. Real-Time High-Performance Semantic Image Segmentation of Urban Street Scenes. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 3258–3274. [CrossRef]
27. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
28. Hao, X.; Hao, X.; Zhang, Y.; Li, Y.; Wu, C. Real-time semantic segmentation with weighted factorized-depthwise convolution. *Image Vis. Comput.* **2021**, *114*, 104269. [CrossRef]
29. Kuntao, C.; Huang, X.; Shao, J. Aggregation Architecture and all-to-one Network for Real-Time Semantic Segmentation. In Proceedings of the ICASSP 2021—2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 2330–2334.
30. Guo, C.; Szemenyei, M.; Yi, Y.; Wang, W.; Chen, B.; Fan, C. Sa-unet: Spatial attention u-net for retinal vessel segmentation. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 1236–1242.
31. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual attention network for scene segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3146–3154.
32. Chen, Y.; Xia, R.; Yang, K.; Zou, K. DARGS: Image inpainting algorithm via deep attention residuals group and semantics. *J. King Saud Univ.-Comput. Inf. Sci.* **2023**, *35*, 101567. [CrossRef]
33. He, J.-Y.; Liang, S.-H.; Wu, X.; Zhao, B.; Zhang, L. Mgseg: Multiple granularity-based real-time semantic segmentation network. *IEEE Trans. Image Process* **2021**, *30*, 7200–7214. [CrossRef]
34. Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J.M.; Luo, P. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 12077–12090.
35. Wan, Q.; Huang, Z.; Lu, J.; Yu, G.; Zhang, L. Seaformer: Squeeze-enhanced axial transformer for mobile semantic segmentation. *arXiv* **2023**, arXiv:2301.13156.
36. Xu, G.; Li, J.; Gao, G.; Lu, H.; Yang, J.; Yue, D. Lightweight real-time semantic segmentation network with efficient transformer and CNN. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 15897–15906. [CrossRef]
37. Zhang, X.L.; Du, B.C.; Luo, Z.C.; Ma, K. Lightweight and efficient asymmetric network design for real-time semantic segmentation. *Appl. Intell.* **2022**, *52*, 564–579. [CrossRef]
38. Bhattacharyya, D.; Thirupathi Rao, N.; Joshua, E.S.N.; Hu, Y.C. A bi-directional deep learning architecture for lung nodule semantic segmentation. *Vis. Comput. B* **2023**, *39*, 5245–5261. [CrossRef]
39. Zhang, X.; Du, B.; Wu, Z.; Wan, T. LAANet: Lightweight attention-guided asymmetric network for real-time semantic segmentation. *Neural Comput. Appl.* **2022**, *34*, 3573–3587. [CrossRef]
40. Wang, J.; Xiong, H.; Nian, X. ADSCNet: Asymmetric depthwise separable convolution for semantic segmentation in real-time. *Appl. Intell.* **2021**, *50*, 1045–1056. [CrossRef]
41. Li, G.; Li, L.; Zhang, J. BiAttnNet: Bilateral attention for improving real-time semantic segmentation. *IEEE Signal Process. Lett.* **2021**, *29*, 46–50. [CrossRef]
42. Gao, G.; Xu, G.; Li, J.; Yu, Y.; Lu, H.; Yang, J. FBSNet: A fast bilateral symmetrical network for real-time semantic segmentation. *IEEE Trans. Multimed.* **2022**, *25*, 3273–3283. [CrossRef]

43. Yan, Q.; Li, S.; Liu, C.; Liu, M.; Chen, Q. RoboSeg: Real-Time Semantic Segmentation on Computationally Constrained Robots. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 1567–1577. [CrossRef]
44. Weng, X.; Yan, Y.; Dong, G.; Shu, C.; Wang, B.; Wang, H.; Zhang, J. Deep multi-branch aggregation network for real-time semantic segmentation in street scenes. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 17224–17240. [CrossRef]
45. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Houlsby, N. An image is worth 16 × 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
46. Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Zhang, L. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 6881–6890.
47. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015; pp. 234–241.
48. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
49. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
50. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
51. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
52. Lu, Z.; Li, J.; Liu, H.; Huang, C.; Zhang, L.; Zeng, T. Transformer for single image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 457–466.
53. Roy, A.G.; Navab, N.; Wachinger, C. Concurrent spatial and channel 'squeeze & excitation'in fully convolutional networks. In Proceedings of the Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, 16–20 September 2018; pp. 421–429.
54. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
55. Chen, Y.; Xia, R.; Zou, K.; Yang, K. FFTI: Image inpainting algorithm via features fusion and two-steps inpainting. *J. Vis. Commun. Image Represent.* **2023**, *91*, 103776. [CrossRef]
56. Zhou, L.; Gong, C.; Liu, Z.; Fu, K. SAL: Selection and attention losses for weakly supervised semantic segmentation. *IEEE Trans. Multimed.* **2020**, *23*, 1035–1048. [CrossRef]
57. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3213–3223.
58. Brostow, G.J.; Shotton, J.; Fauqueur, J.; Cipolla, R. Segmentation and recognition using structure from motion point clouds. In Proceedings of the Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, 12–18 October 2008; pp. 44–57.
59. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
60. Shrivastava, A.; Gupta, A.; Girshick, R. Training region-based object detectors with online hard example mining. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 761–769.
61. Paszke, A.; Chaurasia, A.; Kim, S.; Culurciello, E. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv* **2016**, arXiv:1606.02147.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.