Feature Information Fusion And Lightweight ResNet For Image Semantic Segmentation

Lin Teng[1], Yulong Qiao[1], Yang Sun[2*], and Hang Li[2*]

[1] School of Information and Communication Engineering, Harbin Engineering University, 150001 China
[2] College of Artificial Intelligence, Shenyang Normal University, 110034 China
[*] Corresponding author. E-mail: lihangsoft@163.com

Received: Apr. 05, 2026; Accepted: May. 01, 2026

We introduce an efficient segmentation framework that integrates multi-scale cues in real time, avoiding the need for overly deep architectures. A Separable Pyramid Module (SPM) is introduced to harvest rich context at 1/4 and 1/8 resolutions by combining depthwise-separable, factorized and dilated convolutions in a bottleneck layout, cutting parameters while preserving receptive fields. To guide the fusion of high-level semantics into low-level detail, a Context Channel Attention (CCA) block is proposed. It re-weights shallow feature channels by exploiting the inter-channel correlations learned from deep feature maps, refining edges without extra heavy computation. The overall encoder-decoder is deliberately kept shallow, so that the deepest feature map remains at 1/8 scale, ensuring fast inference. Extensive experiments on PASCAL VOC2012 demonstrate that the new method achieves competitive accuracy against deeper counterparts while maintaining superior speed, validating the effectiveness of the SPM and CCA designs for balancing precision and real-time performance in semantic segmentation tasks.

Keywords: Image semantic segmentation, feature information fusion, lightweight ResNet, context channel attention

http://dx.doi.org/10.6180/jase.202609_32.059

1. Introduction

Semantic segmentation assigns a category label to every pixel, partitioning the image into semantically meaningful regions [1]. In recent years, due to the excellent performance of convolutional neural networks (CNN) in semantic segmentation tasks, the segmentation quality has significantly improved compared to traditional methods. Therefore, the current researches focus has been placed on the design of convolutional neural network structures [2, 3].

Across a CNN stack the representation of a single object mutates: early feature maps keep fine-grained spatial cues that sharpen predicted borders, whereas later, downsampled maps encode high-level semantics that separate semantic classes. Each pooling step both widens the receptive field and shrinks the spatial grid, so the identical target is expressed at several resolutions simultaneously [4]. Exploiting this natural pyramid, so-called multi-scale features, is critical for accurate segmentation. Therefore, how to build a better network structure and effectively utilize multi-scale feature information has become one of the main issues in the current semantic segmentation field. Currently, researchers have designed many effective network structures to improve the quality of semantic segmentation. These models based on convolutional neural networks can be macroscopically regarded as consisting of two parts: the former part generates abstract features containing position and category information through a basic convolutional network, and the latter part generates the predicted segmentation map by utilizing these features. They can be specifically classified into the following three categories: (a) Fully convolutional network-based models (FCN) [5]. These models adopt an encoder-decoder structure. Due to

multiple downsampling operations in the convolutional network at the front stage, the feature maps obtained from the front stage network possess the required high-level semantic features, but their resolution has decreased. To obtain a prediction map of the same size as the original image, multiple upsampling operations will be conducted in the subsequent network. During the upsampling process, by connecting the shallow feature maps, more detailed information is fused to achieve the purpose of improving the segmentation effect. However, the multiple-stage downsampling operations in the front stage network have already caused the loss of a considerable amount of detailed information. Peng et al. [6] addressed the issue of information loss caused by downsampling by adopting a pooling operation with coordinates (indices). During the maximum pooling process, the network recorded the position of the selected maximum pixel on the feature map, and during de-pooling, it restored the maximum value to the corresponding position based on the recorded coordinates, thereby better restoring the details of image segmentation. Even though there is the combination of shallow feature information in the upsampling process in the subsequent network, it also introduces shallow redundant features, which limits the final segmentation effect. (b) DeepLabv2-based models [7]. These models reduce the pooling operations in the initial basic network, thus maintaining the resolution of the feature maps without excessive reduction, while also preserving the detailed information that would be lost due to the reduction of the feature maps. In the following stage, only the final output of the backbone is forwarded; parallel branches are then spawned to harvest a spectrum of feature maps. Each feature map represents a specific scale of features, and finally, these multi-scale feature maps are fused together to predict the segmentation result. For example, DeepLabv3 obtains feature maps with different receptive fields through dilated convolutions with different expansion rates, and then fuses these feature maps through spatial pyramid pooling. Compared with the methods of the third category below, this type of model does not fully utilize the shallow details. Memon et al. [8] believed that the features extracted by Atrous Spatial Pyramid

Pooling (ASPP) were still not dense enough, and thus proposed the DenseASPP algorithm. This algorithm combines the outputs of each dilated convolution in a denser and more complex connection manner, covering a wide range of semantic information and performing feature extraction; Yu Gen et al. designed an image segmentation method that learns multi-scale features through the shared layer of the main network, further constraining and optimizing the target boundary, and obtaining more accurate segmentation results. (c) GCN-based models [9]. These models specify several layers of feature maps to be fused from the front part of the network. During the process of generating the prediction map in the back part, these layers of feature maps are gradually fused from the deep to the shallow levels, thus balancing the semantic features at the deep level and the detailed features at the shallow level. GCN achieves this by iterative methods, continuously refining the low-resolution prediction map with fine-grained shallow features to generate a high-resolution prediction map. (d) Driven by attention's rise in computer vision, soft attention has been grafted onto semantic segmentation to re-weight features on the fly. Wang et al. [10] introduced DANet, a Dual Attention Network that married channel and spatial perspectives: one module learned inter-channel dependencies, the other captured long-range pixel relations. Their outputs were fused to refine the representation, lifting segmentation accuracy markedly.

Lately, deep-learning segmentation engines have advanced rapidly, showing strong promise for augmented reality, self-driving cars, visual search, and human-computer interfaces [11]. However, when reviewing the above methods and other algorithm models in the recent stage, most of them focus on improving the accuracy of image segmentation, with higher computational costs and memory usage, and the real-time performance of the network cannot be guaranteed. The semantic segmentation algorithm designed in this paper based on separable pyramid units ensures the accuracy of image segmentation while also considering the lightness and real-time performance of the network model. Firstly, to address the problem of the expansion convolution effectively expanding the feature extraction receptive field but having high memory usage, a deep separable convolution structure is adopted to reduce the computational cost, and a bottleneck-style feature pyramid is constructed for multi-scale information processing. Secondly, a tri-order encoding network is constructed, with the maximum downsampling to a feature layer with an input image resolution of $1/8$, ensuring fewer model parameters. Finally, through the context channel attention module, the feature weights of each channel in the deep feature map are calculated, and the features of all channels are weighted to the shallow feature map, ultimately obtaining the shallow feature map with modified channel weights, enhancing the channel dependency of the shallow layer and integrating it with the deep feature map, optimizing the target edge segmentation effect and improving the accuracy of semantic segmentation.

2. Materials and methods

2.1. Separable Pyramid Feature Extraction Unit

When deep learning was still in its infancy, An et al. [12] grafted the Deeper Bottleneck Architecture onto ResNet, trimming compute and parameter overhead to speed convergence, see Fig. 1(a). This residual variant squeezes input channels with a 1×1 convolution, letting the following 3×3 layer work on thinner feature maps and slashing multiplication count; afterwards another 1×1 projection restores the original depth while a skip path preserves a slice of the earlier geometry. Chen et al. [13] proposed a one-dimensional non-bottleneck (Non-bottleneck-1D) structure, as shown in Fig. 1(b). It split the 3×3 kernel into a 3×1 followed by a 1×3 stack, cutting parameters and training time yet leaving the receptive field untouched.

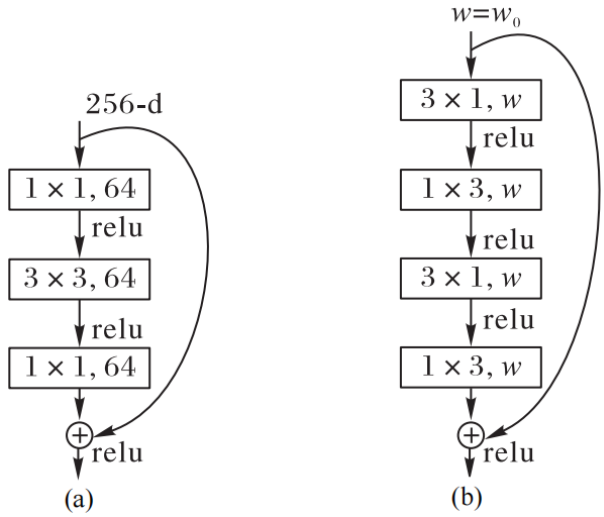

Fig. 1. (a) Deeper bottleneck structure (b) non-bottleneck-1D structure

This paper combines the bottleneck structure in the DBA module and the decomposed convolution to propose the SPM feature extraction unit, as shown in Fig. 2. N represents the number of feature map channels. D represents the depthwise separable convolution. R denotes the dilation rate governing the spacing between kernel weights in dilated convolution. Here, the dilation rates R1, R2, and R3 for different branches are set using an increasing parameter sequence and are processed in parallel for feature extraction. How alternate parameter chains perform will be unpacked in the experiments. Every SPM block first squeezes feature maps to half their depth and later restores it with a point-wise convolution. Although 1×1 kernels introduce fewer weights than 3×3 filters, ResNet's goal is to stack well over a hundred layers, widening the receptive field so richer, more abstract semantics can be captured. But the increase in the number of layers is usually accompanied by more running time and extremely high memory requirements. To construct a lightweight and fast semantic segmentation network model while ensuring the segmentation accuracy, we swap the 1×1 reducer for a 3×3 kernel that halves depth while harvesting broader spatial context; to harvest richer multi-scale cues inside a shallow stack we build the SPM block with split branches whose filters are all replaced by depthwise separable convolution [14], trimming parameters and lifting speed.

In the first branch of this article, a decomposed 3×3 deep convolution is used, which consists of a cascaded 3×1 convolution and a 1×3 convolution. It trims the parameter budget yet keeps the receptive span intact. Taking the feature maps with the same resolution of $H \times W$ as an example, when the convolution kernel size is k and the input channel number is m, the ratio of the number of parameters after applying the deep decomposition convolution to the number of parameters of the depth separable convolution is:

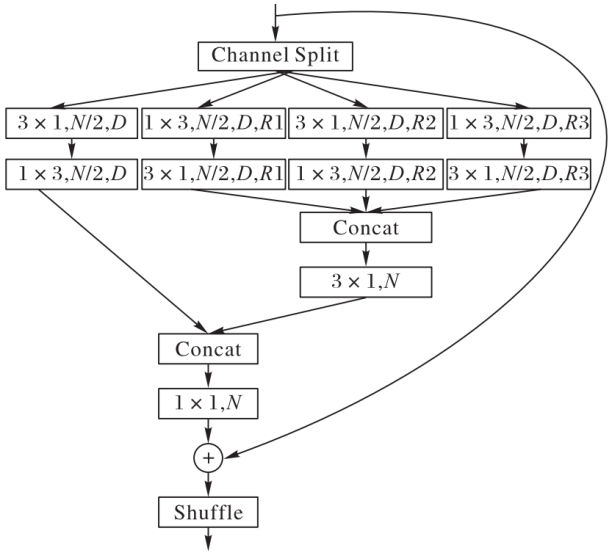

Fig. 2. The proposed SPM feature extraction unit

$$\frac{(k \times 1 \times m \times H \times W) \times 2}{k^2 \times m \times H \times W} = \frac{2}{k} \tag{1}$$

When the kernel size of the convolution is 3×3, the parameter quantity of the depth-decomposition convolution is reduced by 33% compared to the depth-separable convolution. For other 3 branches, dilated convolution is adopted, which expands the receptive field for feature extraction without reducing the resolution of the feature image. Its construction method is also the depth-decomposition convolution. By stacking dilated convolutions with graded

rates the network forms a feature pyramid: wider gaps harvest long-range, intricate context and cost more weights, whereas tighter skips capture short-scale, simple cues with fewer parameters, letting the separable pyramid harvest rich multi-scale context efficiently. Compared with the ASPP of the DeepLab series, the parameters and computational cost of the feature pyramid unit proposed in this paper are significantly reduced. Maps from the pyramid are concatenated, halved by a 1×1 kernel, then merged with the first branch; another point-wise layer boosts channel crosstalk and injects extra non-linearity before the final output. Due to the existence of residual connections, the SPM unit needs to undergo channel random mixing operation again before outputting the final feature map to enhance the information interaction between feature channels.

SPM is designed to harvest ample semantic cues at a shallow level for pixel labeling, delivering strong representation with just a few lightweight layers instead of the 100-plus of ResNet, thus speeding inference while keeping segmentation accuracy.

2.2. Up-and-down channel attention module

In pixel-level segmentation, marrying coarse, category-rich cues from the deepest strata with the fine, location-sensitive signals found near the input is indispensable; yet a naïve concatenation or addition rarely prospers. The uppermost feature maps have been squeezed by successive pooling and striding, so they encode abstract, class-discriminative semantics but have forfeited almost all spatial fidelity. Conversely, early tensors still retain the original resolution, yet their channels are dominated by low-level primitives (edges, corners, gradients and texture noise) that are necessary for delineating object boundaries but powerless for categorical reasoning. Simply mixing the two streams therefore pollutes the reliable semantics with unfiltered noise and yields fuzzy or fragmentary masks.

To escape this dilemma we advocate a guidance paradigm: the high-level layer, already equipped with robust class evidence, should act as a teacher that selectively amplifies or suppresses channels in the low-level student. Concretely, we generate a channel-attention vector from the deepest features, re-weight the shallow feature map in the 1×1 convolutional domain, and only then inject the refined signal into the decoder. In this way the network foregrounds boundary-sensitive channels that coincide with real object contours while muting texture clutter that would otherwise mislead the classifier. Because the whole procedure is implemented with a single global-pooling, fully-connected, sigmoid branch, it introduces next to no extra parameters yet successfully compensates for the semantic poverty of shallow tensors, leading to masks that are both sharply localized and categorically accurate. The modified shallow feature maps, when fused with the deep feature maps, can bring better segmentation results. Therefore, this paper proposes the Context Channel Attention (CCA) module, its structure is shown in Fig. 3.

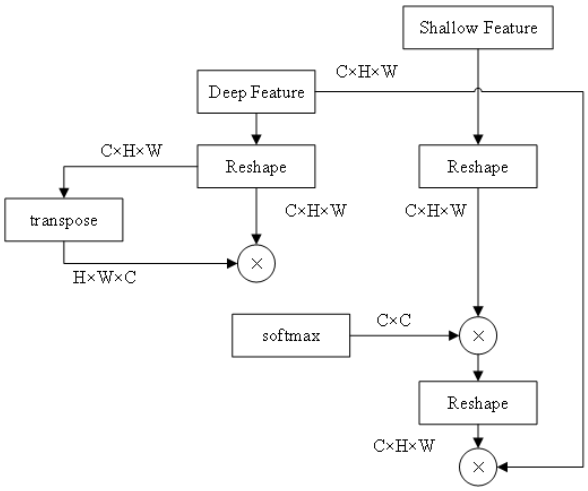

Fig. 3. CCA module

For a deep feature map with dimensions $A \in \mathbb{R}^{C \times H \times W}$, the image is transformed into $\mathcal{R}^{C \times N}(N = H \times W)$ through the matrix dimension transformation operation. Then, it is multiplied by its transpose, and the result is calculated using softmax to obtain the channel attention mapping result $X \in \mathbb{R}^{C \times C}$. Since the internal essence of matrix multiplication is the product of the feature vectors corresponding to the channels with those of all other channels, the connection between channels is successfully modeled through the dot product similarity. The formula is expressed as:

$$x_{ji} = \frac{\exp\left(A_i \cdot A_j\right)}{\sum_{i=1}^{C} \exp\left(A_i \cdot A_j\right)} \tag{2}$$

Here x_{ji} represents the influence of the $i-$ th channel on the $j-$ th channel. $A_i \in \mathbb{R}^{C \times N}$.$A_j$ is the transpose of A_i, $A_j \in \mathbb{R}^{N \times C_i}$. Subsequently, in order to obtain the features of the global dependency, the transpose of the channel feature map is multiplied by the reshaped and dimension-reduced shallow feature map $B_i \in \mathbb{R}^{C \times N}$, which is then reshaped again after multiplying the scale coefficient a to increase the dimension and become the result $\mathbf{R}^{C \times H \times W}$ with the same dimension as the input, that is, the modification result of the channel weights of the shallow feature map under the guidance of the deep layer. The maximum number of channels output by all convolution layers of the proposed network is 128, Hence no squeeze-excitation style

compression (from 512 or 1024 down) is invoked purely to shrink arithmetic cost; once the shallow inter-channel weights are learned they assemble global context and are re-embedded with the incoming deep tensor, yielding a semantically richer output $E \in \mathbb{R}^{C\times H\times W}$ without extra parametric baggage.

$$E_j = \partial \sum_{i=1}^{C} \left(x_j B_i\right) + A_j \tag{3}$$

Since the attention mechanism does not perform well during the initial training, the initial setting is 0 . During the training process, it learns to obtain a larger weight, enabling the CCA module to function. The CCA module calculates the feature weights for each channel of the deep feature map, utilizes the interconnections between all channels, and weights them to the shallow features. Eventually, it obtains the shallow features with modified channel weights, improving the shallow channel dependency. The combination of the guided shallow features with the deep features can better integrate spatial information while maintaining the prediction classification of the encoding part, thereby enhancing the final image segmentation effect.

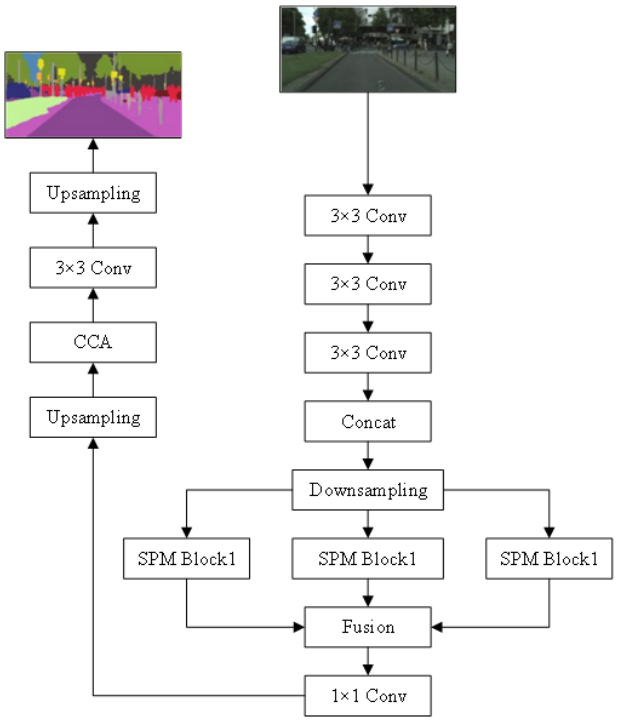

Fig. 4. The proposed network

2.3. Proposed image segmentation structure

In order to strike a better balance between accuracy and real-time performance, this paper only employs three downsampling operations in the proposed network model. The deepest layer obtains feature maps with an initial resolution of 1/8, and the overall network structure is shown in Fig. 4.

This paper first uses three consecutive 3×3 convolutions to extract the initial features of the input image. The first 3×3 convolution has a stride of 2, the second and third 3×3 convolutions have a stride of 1, followed by a downsampling layer that cascades a 3×3 convolution with a stride of 2 and a 2×2 max pooling. The subsequent downsampling layer is a 3×3 convolution with a stride of 2 . Considering that the semantic information contained in the shallow feature maps is not rich and the resolution is high, the benefits of adding SPM units are not sufficient to compensate for the loss caused by the increase in parameters and computational cost. Therefore, this paper does not consider applying SPM units on the feature maps with 1/2 resolution. After the two subsequent downsampling operations, the feature maps with resolutions of 1/4 and 1/8 of the initial input image are input into the SPM Block 1 and SPM Block 2 designed in this paper to extract dense features.

The feature strings extracted from each level are combined together, and the features at all levels are fused through convolution operations. Additionally, for the features of the block layer, additional operations are performed, namely 1×1 convolution operation and global pooling operation. The purpose of adding the former is to retain the original features of the 4 layers, and the purpose of adding the latter is to obtain more global context information.

In summary, for each level of feature maps, not only are the features extracted independently, but also deeper-level features are iteratively used to enrich the semantic information. This is because the more shallow the feature, the more redundant information it usually contains. By merging and integrating with higher-level feature information, more dense feature maps can be generated. The method in this paper can be expressed as follows:

$$f_n = C_n\left(f_{n-1}\right) = C_n\left(C_{n-1}\left(\cdots C_1(I)\right)\right) \tag{4}$$

$$f'_n = f_n, \qquad f'_{n-1} = G_n\left(f'_n, f_{n-1}\right) \tag{5}$$

$$f'_{n-k} = G_{n-k}\left(f'_{n-k+1}, f_{n-k}\right) \tag{6}$$

$$D = \left\{P_{n-k}\left(f'_{n-k}\right), \ldots, P_n\left(f'_n\right)\right\} \tag{7}$$

Eq. (4) represents the basic convolutional network in the previous section, where I represents the input image.

C_n represents the $n-th$ convolutional module, consisting of convolutional layers and pooling layers. f_n represents the feature map of the $n-th$ layer. Eq. (6) represents the fusion process of substituting the deep feature maps into the shallow feature maps, where f_{n-k} is the feature map of the $(n-k)-th$ layer $(0 < k < n)$.

As k approaches n, the represented feature map gets closer to the shallow layer of the network, meaning the resolution of the feature map becomes higher and the detailed information contained therein increases. ${f'}_{n-k}$ represents the fused feature map. G_{n-k} represents fusing the feature map f_{n-k} and the feature map ${f'}_{n-k+1}$ together. From Eq. (6), it can be seen that the feature information of the deeper layers is continuously fused layer by layer with the shallow feature information, enriching the feature information of the shallow layer. Eq. (7) represents the fusion of the obtained multi-level feature maps, P is the refinement operation of the feature map. D represents the fusion operation of all feature maps to obtain the final segmentation result. The detailed parameters of the overall network structure are presented in Table 1.

3. Results and discussion

The dataset used in the experiment is the semantic segmentation part of the public dataset PASCAL VOC2012 [15]. This dataset contains 1464 training images, 1449 validation images, and 1456 test images, including 20 object categories and one background category. To obtain more data, the provided method in reference [16] was utilized to expand the dataset, resulting in 10,582 training images. The network uses the pre-trained ResNet101 [17] on ImageNet as the base model for fine-tuning, and the model performance is evaluated using the mean intersection-over-union (mIoU). Table 2 presents the experimental environment of this study.

The training process iterates a total of 30000 times. During the training, one batch contains 10 images, and the input and output image sizes are both 512×512. At the same time, the data is enhanced by randomly scaling the input images and randomly flipping them left and right. This paper adopts the same method as DeepLabv3+ to control the learning rate (initial learning rate is 0.007). The regularization weight loss is 0.0002 , the loss function is cross-entropy loss, and the optimizer used is Momentum. Additionally, batch normalization is used before each weight layer to simplify the training.

Previously, we introduce the dilated convolutions to extract features at each level. Therefore, in the experiment, we compare the effects of the ordinary 3×3 convolution and dilated convolution on the final segmentation. In addition, considering that the resolution of shallow features is high and there is a lot of redundant information, in order to filter out more redundant information, different dilation rates are used for dilated convolutions at different levels. Table 3 shows the comparison between the ordinary convolution (with a dilation rate of 1) and dilated convolutions with different dilation rates. The dilation rates in the table correspond to the layers from shallow to deep from left to right. From Table 3, it can be seen that as the dilation rate increases, the quality of segmentation gradually improves. However, it reaches its maximum at dilation rates of $21, 15, 7$ and 3, and then begins to decrease. This is because when the dilation rate increases to a certain extent, dilated convolutions become increasingly ineffective. Moreover, the dilation rate of dilated convolutions corresponding to the shallow layers is greater than that of the deep layers, which leads to better results. This also indicates that the shallow features contain more redundant information rates

Although the shallow feature information can precisely segment the objects, the useful feature information extracted from the shallow layer is still less than that extracted from the deep layer. Therefore, when combining the features from different layers, the number of feature map channels for each layer is also different. In the experiments, five sets of channel numbers are adopted respectively. As can be seen from Table 4, when the number of channels in the low-level feature maps is higher and the number of channels in the hierarchical feature maps is lower, the segmentation effect will be better, which also indicates that the effective information content of the low-level features is relatively less.

In order to simply compare the effects of the network itself, the five mainstream methods for comparison are all run in the same experimental environment. Additionally, to better compare the post-processing prediction network, the pre-trained parameters of the initial part of the basic network are fixed and does not participate in the training process.

Table 4 presents the comparison results of the proposed method in this paper with five other advanced methods. The FCN-8s [18] network structure is simple but has difficulty in effectively utilizing shallow information. GCN treats deep features and shallow features equally, and the feature fusion lacks specificity. DenseASPP [19] and DeepLabv3 do not combine shallow feature information. Among these methods, the method in this paper achieved better performance. Compared with the GLDL [20], the number of network layers in this method is less, so the performance is slightly insufficient, but the result difference is

Table 1. Detailed structure of entire network with SPM

Layer	Operation	Parameter	Channel number	Output
1	3 × 3 Conv	stride 2	32	512 × 256
2	3 × 3 Conv		32	512 × 256
3	3 × 3 Conv	stride 1	32	512 × 256
4	Downsample	stride 2	64	256 × 128
5~7	3× SPM-s	stride 1	64	256 × 128
8	Downsample	stride 2	128	128 × 64
9	SPM(3,3,7,7,13,13)	stride 1	128	128 × 64
10	SPM(4,4,8,8,16,16)		128	128 × 64
11	1 × 1 Conv	stride 1	64	128 × 64
12	Upsample	×2	64	256 × 128
13	3 × 3 Conv	stride 1	19	256 × 128
14	Upsample	×4	19	1024 × 512

Table 2. Experimental environment

Environment	**Environment configuration**
System processor	Ubuntu 16.04 LTS Core i7-7800X CPU @ 3.50 GHz ×12
Memory	94 GB
Graphics card	NVIDIA GTX 1080Ti ×2
Deep learning framework	TensorFlow 1.9.0

Table 3. Comparison of ordinary convolution and atrous convolution with different expansion

Expansion rate	mIoU
1,1,1,1	0.6377
3,3,3,3	0.6381
9,7,5,3	0.6406
18,12,6,3	0.6518
21,15,7,3	0.6520
24,18,12,6	0.6371

Table 4. The influence of channel number at each level during the integration process

The channel number for fusing features in each layer	mIoU
8,16,32,64	0.6392
8,24,72,216	0.6397
128,128,128,128	0.6424
16,32,64,128	0.6406
32,64,128,256	0.6520

relatively small, only 0.47%, which is sufficient to show that this method can obtain good semantic segmentation results and verifies the effectiveness of this method. Fig. 5 shows the segmentation results of this method. It can be seen that this new method can achieve the purpose of semantic segmentation.

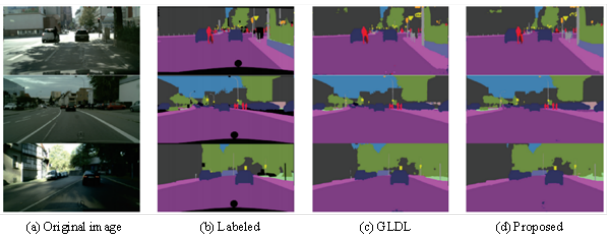

Fig. 5. Visualized segmentation results

4. Conclusions

Motivated by the persistent tension between mask fidelity and frame-rate, this study first dissects the accuracy-versus-latency behaviour of contemporary segmentation models. We then introduce a real-time architecture that departs from heavy encoders and instead relies on separable-pyramid feature-extraction cells; these units assemble a multi-scale receptive field with markedly fewer FLOPs and parameters. To prevent the usual loss of fine detail, a lightweight

context-channel attention gate is deployed: it treats high-level tensors as semantic anchors and re-weights shallow feature maps on-the-fly, so edges and textures that align with object hypotheses are amplified while noisy responses are suppressed. This guided fusion strategy yields crisp boundaries without additional computational bulk. Extensive experiments on Cityscapes, CamVid and our own high-resolution set reveal that the proposed network delivers a superior accuracy-to-speed equilibrium, outperforming a range of established compact designs as well as deeper, slower counterparts. The constructed feature extraction module, attention method, and lightweight network model are of reference significance for other researchers. Since the algorithm in this paper is only deeply tested based on one dataset, it lacks specificity for specific target categories. In subsequent research, the network design will consider combining specific image segmentation targets to further improve the practical performance of the model.

5. Acknowledgements

None

References

[1] S. Yin, L. Wang, T. Chen, H. Huang, J. Gao, J. Zhang, M. Liu, P. Li, and C. Xu, (2026) *"LKAFormer: A lightweight kolmogorov-arnold transformer model for image semantic segmentation"* **ACM Transactions on Intelligent Systems and Technology *17***(3): 1–24. DOI: 10.1145/3759254.

[2] Y. Lu, Y. Chen, D. Zhao, and J. Chen. "Graph-FCN for image semantic segmentation". In: *International symposium on neural networks*. Springer. 2019, 97–105. DOI: 10.1007/978-3-030-22796-8_11.

[3] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, (2018) *"A survey on deep learning techniques for image and video semantic segmentation"* **Applied Soft Computing *70***: 41–65. DOI: 10.1016/j.asoc.2018.05.018.

[4] K. Wang, J. H. Liew, Y. Zou, D. Zhou, and J. Feng. "Panet: Few-shot image semantic segmentation with prototype alignment". In: *proceedings of the IEEE/CVF international conference on computer vision*. 2019, 9197–9206. DOI: 10.1109/ICCV.2019.00929.

[5] W. Sun and R. Wang, (2018) *"Fully convolutional networks for semantic segmentation of very high resolution remotely sensed images combined with DSM"* **IEEE Geoscience and Remote Sensing Letters *15***(3): 474–478. DOI: 10.1109/LGRS.2018.2795531.

[6] C. Peng, Y. Li, L. Jiao, Y. Chen, and R. Shang, (2019) *"Densely based multi-scale and multi-modal fully convolutional networks for high-resolution remote-sensing image semantic segmentation"* **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing *12***(8): 2612–2626. DOI: 10.1109/JSTARS.2019.2906387.

[7] W. Zhao, Y. Chen, S. Xiang, Y. Liu, and C. Wang, (2023) *"Image semantic segmentation algorithm based on improved DeepLabv3+"* **Journal of System Simulation *35***(11): 2333–2344. DOI: 10.16182/j.issn1004731x.joss.22-0690.

[8] M. M. Memon, M. A. Hashmani, A. Z. Junejo, S. S. Rizvi, and K. Raza, (2022) *"Unified DeepLabV3+ for semi-dark image semantic segmentation"* **Sensors *22***(14): 5312. DOI: 10.3390/s22145312.

[9] D. Jiang, H. Qu, J. Zhao, J. Zhao, and W. Liang, (2021) *"Multi-level graph convolutional recurrent neural network for semantic image segmentation"* **Telecommunication Systems *77***(3): 563–576. DOI: 10.1007/s11235-021-00769-y.

[10] W. Wang, S. Wang, Y. Li, and Y. Jin, (2021) *"Adaptive multi-scale dual attention network for semantic segmentation"* **Neurocomputing *460***: 39–49. DOI: 10.1016/j.neucom.2021.06.068.

[11] Z. Wang, Z.-H. You, N. Xu, C. Zhang, and D.-S. Huang, (2024) *"UAVSeg: Dual-encoder cross-scale attention network for UAV images' semantic segmentation"* **IEEE Transactions on Geoscience and Remote Sensing *63***: 1–17. DOI: 10.1109/TGRS.2024.3502401.

[12] Z. An, J. Zhang, Z. Sheng, X. Er, and J. Lv, (2021) *"RBDN: Residual bottleneck dense network for image super-resolution"* **Ieee Access *9***: 103440–103451. DOI: 10.1109/ACCESS.2021.3096548.

[13] L. Chen, X. Xu, L. Pan, J. Cao, and X. Li, (2021) *"Real-time lane detection model based on non bottleneck skip residual connections and attention pyramids"* **Plos one *16***(10): e0252755. DOI: 10.1371/journal.pone.0252755.

[14] L. Teng, Y. Qiao, M. Shafiq, G. Srivastava, A. R. Javed, T. R. Gadekallu, and S. Yin, (2023) *"FLPK-BiSeNet: Federated learning based on priori knowledge and bilateral segmentation network for image edge extraction"* **IEEE Transactions on Network and Service Management *20***(2): 1529–1542. DOI: 10.1109/TNSM.2023.3273991.

[15] X. Li, X. Yong, T. Li, Y. Tong, H. Gao, X. Wang, Z. Xu, Y. Fang, Q. You, and X. Lyu, (2024) *"A spectral–spatial context-boosted network for semantic segmentation of remote sensing images"* **Remote Sensing *16***(7): 1214. DOI: 10.3390/rs16071214.

[16] M. Luo, Y. Zan, K. Khoshelham, and S. Ji, (2025) *"Domain generalization for semantic segmentation of remote sensing images via vision foundation model fine-tuning"* **ISPRS Journal of Photogrammetry and Remote Sensing *230***: 126–146. DOI: 10.1016/j.isprsjprs.2025.09.004.

[17] L. Wang, D. Li, S. Dong, X. Meng, X. Zhang, and D. Hong, (2025) *"PyramidMamba: Rethinking pyramid feature fusion with selective space state model for semantic segmentation of remote sensing imagery"* **International Journal of Applied Earth Observation and Geoinformation *144***: 104884. DOI: 10.1016/j.jag.2025.104884.

[18] J. Liu, H. Zhang, J. Chen, R. Meng, C. Gao, L. Han, Y. Song, Y. Tian, and Y. Wang, (2025) *"Automated detection and segmentation of dental caries using a novel cascaded learning approach"* **Biomedical Signal Processing and Control *102***: 107344. DOI: 10.1016/j.bspc.2024.107344.

[19] Z. Rui, L. Han, Y. Li, and G. Liu. "Semantic segmentation of building floor plans based on improved Deeplabv3+". In: *Second International Conference on Remote Sensing Technology and Survey Mapping (RSTSM 2025)*. ***13802***. SPIE. 2025, 301–307. DOI: 10.1117/12.3067853.

[20] Z. Wu, M. Li, Y. Han, and X. Feng, (2025) *"Semantic segmentation of 3D point cloud for sewer defect detection using an integrated global and local deep learning network"* **Measurement *253***: 117434. DOI: 10.1016/j.measurement.2025.117434.