Academic Journal of Science and Technology
ISSN: 2771-3032 | Vol. 14, No. 3, 2025

Research on Semantic Segmentation Algorithms for Street Scenes

Linlin Liu*

School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, Henan 45400, China
*Corresponding Author: Linlin Liu

Abstract: Some advanced semantic segmentation models often feature deep network structures and a large number of parameters, requiring significant memory and computational resources. This makes them difficult to run in real-time on devices with limited computational capacity. The model may fail to complete the semantic segmentation task within the required timeframe, while the complex feature extraction process further increases computational demands and processing time, reducing real-time performance. This, in turn, slows system response speed and negatively impacts practical applications. To address the trade-off between segmentation accuracy and real-time performance, this paper proposes an up-sampling module-the Scale-Aware Depth wise Separable Convolution Attention Module (SADAM). This module enhances the model's ability to focus on key regions, improving feature extraction efficiency while reducing model complexity and boosting inference speed without compromising segmentation accuracy. Additionally, a semantic-assisted optimization branch is introduced to incorporate more feature information, enhancing the model's representation capacity and adaptability. Furthermore, the loss function is optimized to improve segmentation accuracy and completeness.

Keywords: Deep learning; Semantic segmentation; Upsampling module.

1. Introduce

Semantic segmentation is a computer vision technique for pixel-wise classification, aiming to achieve fine-grained scene analysis and structured understanding by assigning each pixel in an image to a specific semantic category (e.g., "vehicle," "pedestrian," "road"). Unlike object detection, which primarily focuses on localizing objects, semantic segmentation requires algorithms to differentiate semantic regions at the pixel level while ensuring spatial consistency within the same category.

Deep learning-based semantic segmentation methods are typically built upon Convolutional Neural Networks (CNNs)[1]. Compared to traditional classification networks, their key innovation lies in adopting a fully convolutional design, where convolutional layers replace fully connected layers as the classification output, preserving the spatial dimension of feature maps. Modern segmentation models commonly employ an encoder-decoder structure[2].The encoder progressively reduces the feature map resolution through multiple convolutional and pooling layers, extracting high-dimensional semantic representations. The decoder restores the feature map resolution using transposed convolutions (deconvolutions) or upsampling techniques, ultimately producing a pixel-wise classification output that matches the input image dimensions. To mitigate detail loss during downsampling, classic models like U-Net introduce skip connections, fusing shallow encoder features with deep decoder features to enhance edge segmentation accuracy.

It is essential to distinguish semantic segmentation from instance segmentation[3]. While semantic segmentation focuses solely on categorizing pixels into general classes (e.g., all "pedestrian" pixels belong to the same category), instance segmentation further differentiates individual objects within the same category (e.g., assigning distinct labels to each pedestrian). This distinction determines their respective applications. Instance segmentation is crucial for scenarios requiring precise object counting, such as retail foot traffic analysis. Semantic segmentation is better suited for global environmental understanding, such as autonomous driving (for drivable area segmentation) or medical imaging (for organ and lesion localization).Today, semantic segmentation has been widely deployed across multiple domains. Autonomous driving: Enables real-time analysis of roads, obstacles, and traffic signs for environmental perception. Medical diagnostics: Accurately segments tumor regions in CT/MRI scans, assisting in pathological analysis. Remote sensing: Facilitates land cover classification and disaster assessment using aerial and satellite imagery.

2. Current Research Status at Home and Abroad

2.1. Methods of Semantic Segmentation

2.1.1. Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are a type of deep learning model widely used in fields such as image recognition and speech recognition. The core idea of CNNs is to extract features from data (such as images) through convolution operations, enabling tasks such as classification and recognition.The basic structure of a CNN consists of convolutional layers, pooling layers, and fully connected layers:The convolutional layer is the core component of CNNs. It applies a sliding convolution kernel over the input data to perform convolution operations, extracting local features. The convolution operation can be seen as a special weighted summation, where the weights in the convolution kernel are learned through training.The pooling layer is used to downsample the feature maps produced by the convolutional layer, reducing the number of model parameters and computational complexity. Common pooling methods include max pooling and average pooling.The fully connected layer maps the output feature vectors from the pooling layer to the final output categories, enabling

classification or recognition of the input data.

CNNs exhibit translation invariance and parameter sharing, which enhance the generalization ability and robustness of the model. In summary, Convolutional Neural Networks are deep learning models based on convolution operations, which automatically learn and extract features from data to perform classification, recognition, and other tasks.

2.1.2. Fully Convolutional Networks(FCNs)

Fully Convolutional Network (FCN) [4] is a crucial architecture in deep learning for image processing tasks. Compared to traditional CNNs, the key feature of FCN is that its output layer is a dense pixel-wise feature map, where each pixel corresponds to a local receptive field in the input image. This enables FCN to not only identify objects in an image but also perform pixel-level predictions, such as image segmentation and image generation, as illustrated in **Figure 2-1**.

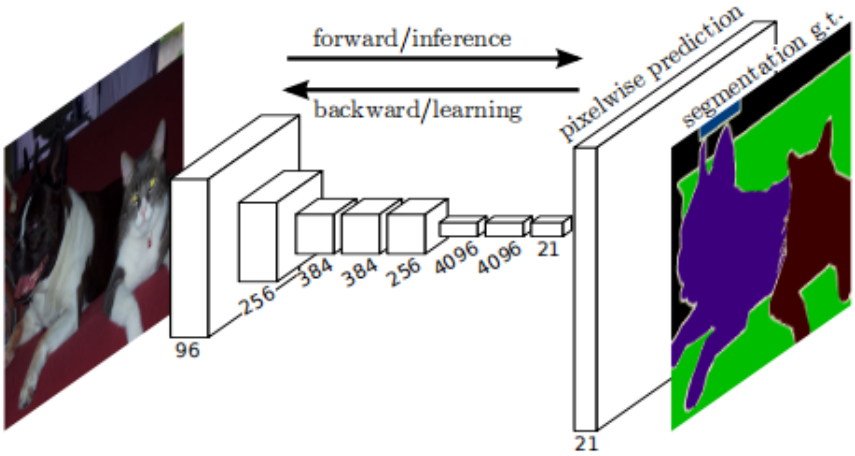

Figure 2-1. FCN Model Diagram

2.2. Deep Learning-Based Semantic Segmentation

Each architecture has subtle differences that distinguish it from standard models, giving it unique advantages when applied to specific problems. These architectures fall under the category of "deep" models, which often outperform shallow models in terms of performance.

With the advancement of machine learning, numerous deep learning architectures have been proposed. AlexNet, introduced by Geoffrey Hinton and his colleagues[5], was the first deep architecture. It is a simple yet powerful neural network that laid the foundation for modern deep learning. VGGNet[6], developed by the Visual Geometry Group (VGG) at the University of Oxford, is characterized by a pyramid-like structure, where the lower layers are wide, and the upper layers are narrow and deep.GoogleNet (also known as InceptionNet) was designed by Google researchers[7]. This powerful model not only increased network depth but also introduced a novel method called the Inception module.ResNet[8] (Residual Network) is one of the architectures that truly defined deep learning models. It consists of multiple residual blocks, which form the building blocks of ResNet architectures and help train deeper networks effectively. YOLO (You Only Look Once)[9] is an advanced real-time system built on deep learning principles, designed specifically to solve object detection problems.SegNet[10] is a deep learning architecture developed for image segmentation. It consists of a series of encoding layers (encoder) followed by corresponding decoding layers, enabling pixel-level classification.

The continuous development of neural network architectures has brought significant value to modern information technology and communication fields. It has not only accelerated digital transformation in businesses but has also had a profound impact on personal life, education, healthcare, and various other sectors.

3. Related Work

3.1. PIDNet

The network structure of PIDNet[11] is illustrated in **Figure 3-1**. The input image undergoes three convolutional downsampling operations, producing a feature map at 1/8 of the original image resolution, which serves as the input for the three branches of PIDNet.

Proportion (P) Branch: This branch is responsible for analyzing and preserving detailed information in high-resolution feature maps. Both the input and output feature maps maintain a 1/8 resolution of the original image, with channel dimensions stacked from 64 to 128. The Pag module enables interaction between the P branch and the Integration (I) branch, while also passing output to the S-Head for network loss calculation. Integration (I) Branch: This branch is designed to aggregate both local and global contextual information to capture long-range dependencies. It starts with a 1/8 resolution feature map and undergoes three downsampling operations, eventually reaching 1/64 of the original resolution. During the second and third downsampling stages, interactions with the P branch occur via the Pag module. The P and I branch inputs undergo element-wise multiplication, followed by attention-based selective learning, constraining values between 0 and 1.Parallel Aggregation Pyramid Pooling Module (PAPPM): Similar to Feature Pyramid Network (FPN), PAPPM fuses multi-scale features through bottom-up and lateral connections, enhancing semantic feature representation. Differentiation (D) Branch: This branch extracts high-frequency features to predict boundary regions. Information from the D branch is fused with the I branch through feature map summation.

Finally, the outputs from the three PID branches are fused using the Bag attention-guided module, producing the final

network output with the channel dimension compressed to 128.

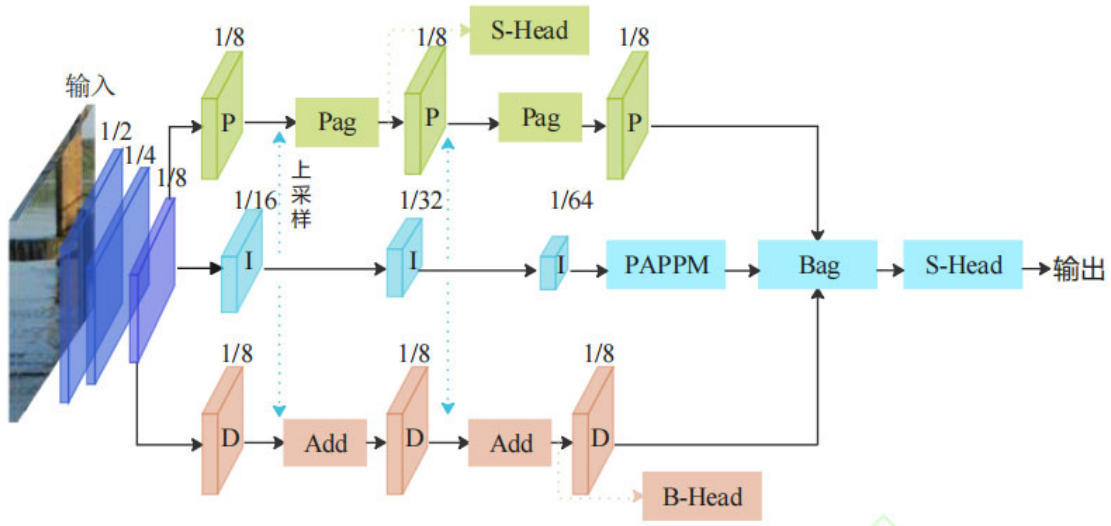

Figure 3-1. PIDNet Architecture Diagram

3.2. Efficient Upsampling Module (EUCB)

As an Efficient Up-Convolution Block (EUCB)[12], EUCB is designed to progressively upsample feature maps, aligning their size and resolution with subsequent skip connections. This alignment enhances information fusion across different layers and stages, making it particularly effective for segmentation networks.

The EUCB operates as follows:Upsampling Operation: The input feature map is scaled up by a factor of 2.Depthwise Convolution (DWC): Applied after upsampling to extract spatial features.Batch Normalization (BN) & ReLU Activation: These operations efficiently enhance the feature map without significantly increasing computational cost.1×1 Convolution: Reduces the number of channels in the upsampled feature map to match the channel dimensions of the next stage.This channel alignment is crucial for smooth integration in the decoder path, ensuring effective feature propagation during semantic segmentation.

4. Proposed Algorithm

4.1. Semantic-Assisted Optimization Branch

In the PIDNet network architecture, a novel three-branch structure is employed, consisting of a detail branch, a semantic branch, and a boundary branch. The boundary branch guides the fusion of the detail and semantic branches. When the feature map enters the model, it first undergoes feature extraction through the semantic branch, where three downsampling operations reduce the feature map to 1/8 of the original image size. However, this process may result in information loss.

To address this limitation, an additional Semantic-Assisted Optimization Branch (referred to as the "fourth branch") is introduced. This branch enhances the model's performance as follows: When the feature map is downsampled to 1/4 of the original size, it is processed through the Efficient Up-Convolution Block (EUCB) to increase the number of channels. More channels allow the model to capture richer feature information, improving representation capability and accuracy. The feature map is then downsampled to 1/8 using the F.interpolate() function and element-wise added to the original 1/8 resolution feature map. The resulting feature map is further processed through another EUCB module to increase the number of channels again and undergoes another downsampling using F.interpolate().When the feature map reaches 1/64 of the original image size, it is fed into the Boundary-Attention-Guided Fusion Module (Bag), which introduces additional feature information to further enhance the model's representation capability and adaptability.

4.2. SADAM

In semantic segmentation, an upsampling module is used to restore low-resolution feature maps back to the same size as the original input image. This process is primarily employed in the decoder stage of deep learning models, particularly Convolutional Neural Networks (CNNs), with the goal of recovering spatial resolution to enable pixel-wise classification.

Semantic segmentation requires the model to classify each pixel in an image. Typically, deep networks extract high-level features through multiple layers of convolution and pooling operations, but this also results in a loss of spatial resolution. To address this issue, this paper introduces an upsampling module called the Scale-Aware Depthwise Separable Convolution Attention Module (SADAM).The workflow of SADAM is as follows: Feature Extraction: Receives an RGB input image (assumed to have dimensions H×W×3).The input passes through an initial convolution layer (Conv) for feature extraction, producing an output feature map of size H×W×64 with 64 channels. Core Module – Multi-Branch Depthwise Separable Convolution (DWC):The feature map is split into two branches for parallel processing: Left Branch: Depthwise Separable Convolutions (DWC) with dilation rates d = {1, 2, 3}, capturing multi-scale contextual information. A 1×1 convolution is applied to adjust the channel dimensions, maintaining 64 output channels per branch. The feature maps are bilinearly upsampled to restore their spatial resolution (H×W).Right Branch: Global Average Pooling (GAP) is used to compress spatial information, reducing the spatial dimensions to 1×1×64.A fully connected layer (FC) followed by non-linear activation functions (e.g., ReLU/Sigmoid) generates channel attention weights. These weights are

applied to channel-wise scale the left branch features. Feature Fusion & Attention Map Generation: The left and right branch outputs are element-wise multiplied to fuse the features. The result is passed through a Sigmoid activation function, producing the final attention heatmap of size H×W×1, with values normalized in the range [0,1].

This entire process emphasizes multi-scale feature fusion and dynamic channel attention adjustment, making it highly suitable for handling complex semantic segmentation tasks. The SADAM model structure is illustrated in **Figure 4-1**.

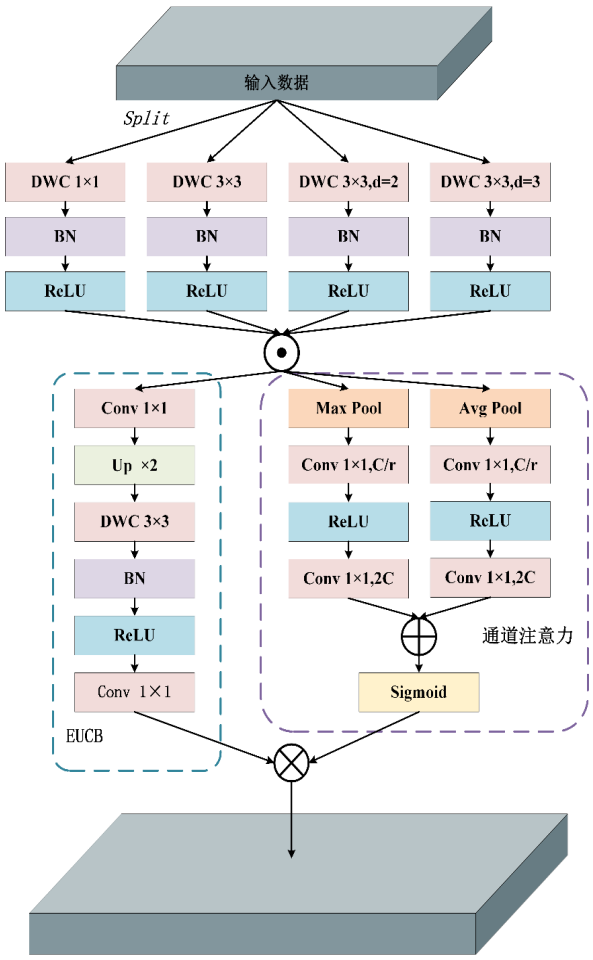

Figure 4-1. SADAM Model Diagram

4.3. Loss Function

In deep learning, the loss function is crucial for model training. In the PIDNet network, OhemCrossEntropy or CrossEntropy is selected as the base loss function. The former accelerates training by selecting difficult samples during the training process, while the latter calculates cross-entropy loss for all samples without distinguishing between difficult and simple samples. This paper designs a combined loss function, CombinedLoss, which combines multiple loss functions and controls the weight of each loss function in the final loss calculation. With alpha set to 0.7, OhemCrossEntropy contributes 70% to the loss. The goal is to leverage different loss information to enhance the model's training performance while optimizing pixel-level accuracy and shape similarity.

5. Conclusion

This paper focuses on the optimization and improvement of PIDNet. First, a scale-aware depth-separable convolution attention module (SADAM) is designed as an upsampling module. Through three parallel depth-separable convolution branches (with dilation rates d=1, 2, and 3), the module captures features of local details (d=1), medium-range (d=2), and global context (d=3). After adding the features from each branch, a 1×1 convolution is applied to adjust the channel number, ensuring effective integration of multi-scale information and maintaining consistency with the original input size to avoid information loss. The internal branches extract channel-level information through global average pooling (GAP) and use a compression-expansion structure in the fully connected layer to generate channel attention weights. Sigmoid outputs the activation strength of each channel, suppressing redundant features and enhancing key channels. The separation of spatial convolution and channel projection reduces about 90% of the computational cost compared to standard convolution, maintaining the lightweight characteristics of the overall module. Then, a "fourth branch" (semantic-assisted optimization branch) is proposed to solve the problem of feature loss during the downsampling process in the original PIDNet. Finally, a combined loss function, CombinedLoss, is designed, combining OhemCrossEntropy (70%) and CrossEntropy (30%) to enhance the learning ability of difficult samples, optimizing pixel-level accuracy and shape similarity. In conclusion, through the proposal of a novel upsampling module, optimization of PIDNet's network structure, design of a combined loss function, and thorough experimental validation, a high-precision, low computational cost, and strong generalization ability semantic segmentation model is achieved. These improvements not only enhance the overall performance of PIDNet but also provide a reference for future lightweight and efficient segmentation networks.

References

[1] Li Z, Liu F, Yang W, et al. A survey of convolutional neural networks: analysis, applications, and prospects[J]. IEEE transactions on neural networks and learning systems, 2021, 33(12): 6999-7019.

[2] Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 39(12): 2481-2495.

[3] Guo Y, Liu Y, Georgiou T, et al. A review of semantic segmentation using deep neural networks[J]. International journal of multimedia information retrieval, 2018, 7: 87-93.

[4] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.

[5] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Advances in neural information processing systems, 2012, 25.

[6] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[7] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.

[8] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[9] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the

IEEE conference on computer vision and pattern recognition. 2016: 779-788.

[10] Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 39(12): 2481-2495.

[11] Xu J, Xiong Z, Bhattacharyya S P. PIDNet: A real-time semantic segmentation network inspired by PID controllers[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023: 19529-19539.

[12] Rahman M M, Munir M, Marculescu R. Emcad: Efficient multi-scale convolutional attention decoding for medical image segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024: 11769-11779.