EPSSNet:
A Lightweight Network With Edge Processing and Semantic Segmentation for Mobile Robotics

Zechun Cao, Texas A&M University, San Antonio, USA*

German Zavala Villafuerte, Texas A&M University, San Antonio, USA

Joseph Almaznaai, Texas A&M University, San Antonio, USA

ABSTRACT

Fast and accurate segmentation is important for robot judgement, e.g. robot detection, segmentation, and control. Most researchers have focused on deploying lightweight semantic segmentation models into robot services. The problem is that the critical interaction between semantic segmentation and boundaries is ignored. In this chapter, the authors propose a lightweight parallel execution model (EPSSNet) based on semantic flow branch (SFB), edge flow branch (EFB) and self-adapting weighting fusion (SAWF) for mobile robot service projects. The semantic flow branching module is used to obtain accurate object shape features. The boundary constraint module uses multiple convolution and upsampling to distinguish boundary features from semantic features. In order to adaptively fuse boundary features with semantic segmentation features, the SAWF is proposed. It adaptively fuses semantic and boundary features by learning boundary and semantic feature fusion weights. Detailed experimental results on Cityscapes, Pascal VOC 2012 and ADE20k datasets demonstrate the superior performance of our approach.

KEYWORDS

Boundary, Constraint, EFB, EPSSNet, Robotic Service, SFB, SAWF

INTRODUCTION

In recent years, rapid advancements in machine learning and deep learning have found extensive applications across various domains. For instance, supervised classification leveraging machine learning techniques has been explored (Salhi et al., 2021). Text analysis has notably benefited from deep learning methodologies (Singh & Sachan, 2021; Ismail et al., 2022; Gu et al., 2022), alongside sentiment analysis (Mohammed et al., 2022), industrial applications (Sharma et al., 2022), medical diagnostics (Xu et al., 2021), disease safety detection (Nguyen et al., 2021), and image enhancement tasks like defogging (Liu et al., 2022). The metaverse (Deveci et al., 2022) emerges as a groundbreaking platform for experimenting with autonomous driving, heavily reliant on deep learning for its core technology. Image segmentation and boundary detection are crucial in the field of computer vision,

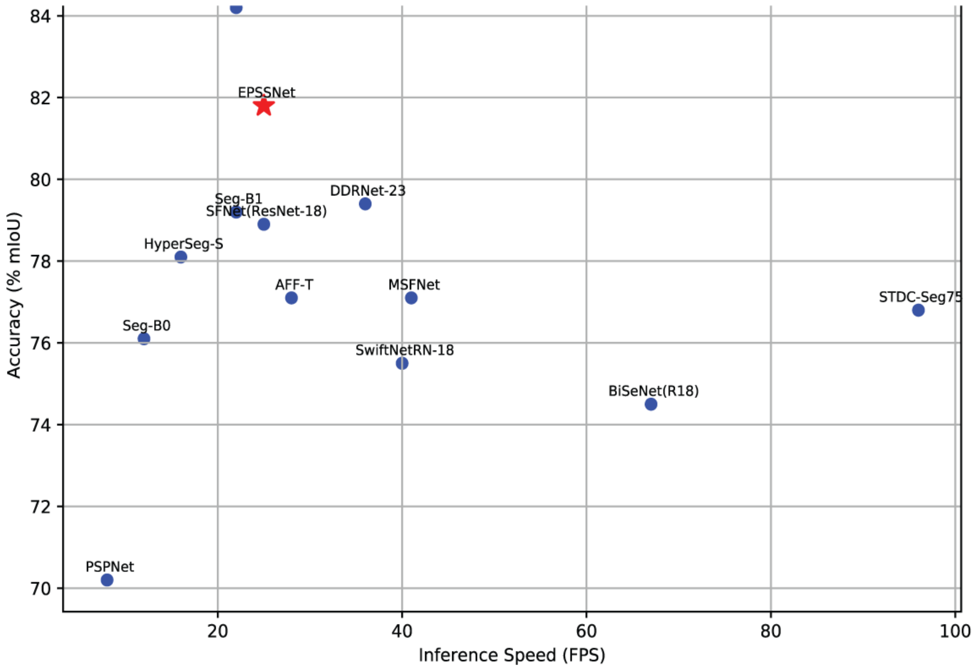
DOI: 10.4018/IJSWIS.342087

*Corresponding Author

serving various fields such as autonomous driving assistance (Teichmann et al., 2018), simultaneous localization and mapping (SLAM) (Chen et al., 2021), point cloud segmentation (Wang et al., 2022), and medical imaging (Liu et al., 2022). Semantic segmentation involves assigning specific labels to individual pixels within objects, while boundary detection focuses on delineating object edges. However, prevalent neural network architectures like FCN (Long et al., 2015), ODE (Zhou et al., 2014), and UERF (Luo et al., 2016) face challenges in effectively capturing extensive pixel relationships as network depth increases, impeding accurate pixel classification. Moreover, deeper networks introduce noise and interference, further complicating the precise classification of minimal pixel clusters and leading to resolution loss and blurring during feature extraction downsampling. As network depth increases, external factors increasingly interfere with end-to-end segmentation. Predicted image outputs often contain unknown pixel classes, significantly impacting subject boundary segmentation quality. Accurate recognition of subject boundary pixels is crucial, especially for mobile robots operating in various environments. The main focus of boundary detection in the study aims to precisely locate subject boundary pixels, even in scenarios involving multiple object classes, where edge pixels belonging to different classes can lead to inadequate environmental understanding by mobile robots.

To overcome this problem, JSBD (Zhen et al., 2020) proposed that the appearance of the subject often comes with a boundary, and the appearance of the boundary contour also refines the subject, so both advantages can be utilized to guide each other's learning. Ada-detector (Sun et al., 2022) proposed the RRT boundary detection algorithm, which enhances the speed of boundary detection by restricting the region detected by RRT to the vicinity of the boundary point. HFDS (Xu et al., 2021) uses global exploration and local exploration trees, which ensures that the robot detects the unknown region, reduces the repeated detection of the robot, and improves efficiency. In addition, the subject boundary can localize the object's relative pose and position, and the semantic subject

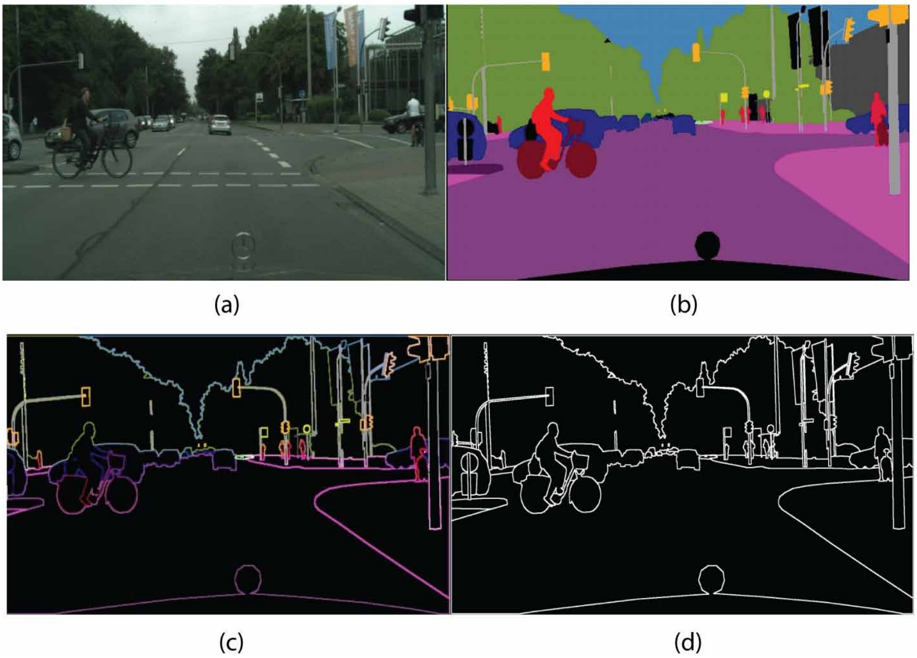
Figure 1. EPSSNet is compared with other lightweight models



is very important for understanding the object. However, numerous existing approaches, including SegFormer (Xie et al., 2021), TopFormer (Zhang et al., 2022), Mobilevit (Mehta & Rastegari, 2021), and Mobileformer (Liu et al., 2021), primarily emphasize crafting lightweight models to improve the real-time performance of semantic segmentation while overlooking boundary accuracy. Moreover, ISSDBES (Li et al., 2020) employs flow field and twisting algorithms to enhance performance, yet the flow field is vulnerable to noise interference, resulting in imprecise boundary segmentation. GSCNN (Takikawa et al., 2019) introduces a dual-path network and refines semantic and boundary masks using dual-task loss but overlooks the associated computational burden. In summary, although considerable efforts have been made to develop lightweight models, pursuing a model capable of simultaneously capturing semantic context and edge details has not received adequate attention. This capability enables mobile robots to perceive their surroundings swiftly and accurately. Consequently, addressing these challenges and requirements, we explore the design of a lightweight model achieved through the effective weighted fusion of semantic and border feature maps with original images, semantic feature maps, border feature maps, and real labels, as illustrated in Figure 2.

Here, we introduce a lightweight framework that integrates border information and semantic content concurrently. Presently, to implement the designed semantic segmentation model for mobile robots and autonomous driving, KSF-SLAM (Zhao et al., 2022) introduces a strategy for key-frame-selective segmentation, enhancing the real-time performance of SLAM models. TIR (Dosovitskiy et al., 2021) proposes a potent lightweight transformer framework that effectively reduces computational costs. Moreover, numerous lightweight attention mechanisms have recently emerged to capture long-range contextual relationships, such as self-attention (Vaswani et al., 2017), channel-attention (Hu et al., 2018), soft-attention (Li et al., 2019), and aggregate-attention (Shi, 2023). These mechanisms establish long-range contextual dependencies, capture inter-pixel relationships, and maintain low computational costs. However, they come with drawbacks:

Figure 2. Effective weighted fusion of semantic and border feature maps with original images, semantic feature maps, border feature maps, and real labels (Note. From left to right, top to bottom: (a) original image, (b) semantic segmentation labels, (c) boundary features, and (d) real labels)



- high data requirements: Often demanding substantial data and computational resources.
- poor interpretability: The complex internal structure of attention mechanisms complicates fine-tuning for desired effects.
- difficulty capturing fine-grained information
- simplistic decoder design: Many attention-based Transformers feature simplified decoder designs, hindering detailed information recovery. Some models, like Afformer (Bo et al., 2023) and Seaformer (Wan et al., 2023), even eliminate the decoder design to alleviate model burdens, worsening boundary blurring.

Transformer-CNN (Tablib et al., 2024) proposes combining transformer's attention mechanism with CNN feature extraction capabilities for more accurate detection results. However, the model entails high computational complexity, making training and inference processes time-consuming. LA-transformer (Caron et al., 2024) combines position-aware mechanisms and self-supervised learning to enhance semantic segmentation accuracy and generalization but suffers from high computational complexity and weak generalization on uneven data.

DT-transformer (Liu et al., 2024) introduces the dynamic token-pass mechanism to enhance semantic understanding at different image locations, but at the expense of increased computational complexity. U-MixFormer (Yemo et al., 2023) combines UNet structure and transformer's mix-attention mechanism to reduce computational complexity and improve efficiency, albeit with limited generalization ability. P2AT (Elhassan et al., 2023) combines pyramid pooling and axial transformer to enhance perceptual range and accuracy, yet with significant computational complexity and limited generalization ability.

- SEDDDS (Liu et al., 2022) introduces a multitasking framework to address semantic edge problems but overlooks computational efficiency. Similarly, RCFED (Liu et al., 2017) utilizes advanced convolutional features to enhance edge refinement, neglecting computational performance. BASeg (Xiao et al., 2023) combines boundary detection and semantic segmentation by leveraging boundary information to guide context aggregation, yet lacks a lightweight design. Similarly, SegFix (Yuan et al., 2020) replaces boundary pixels with internal pixels segmented by the semantic subject, albeit as a model-independent post-processing mechanism, thereby increasing the model burden. Although BASeg and SegFix yield superior results, their lack of lightweight design precludes deployment in mobile robots.
- To overcome these constraints, we propose a lightweight model, EPSSNet, explicitly designed for mobile robots, integrating semantic segmentation and boundary constraints. EPSSNet comprises a Semantic Flow Module (SFB) and a Boundary Flow Module (EFB), enabling simultaneous semantic segmentation and boundary detection while facilitating mutual learning. Moreover, we introduce a self-adaptive fusion module (SAWF) following semantic and boundary flow paths. SAWF learns adaptive weights for boundary and semantic feature maps simultaneously, allowing weighted clustering of same-class features and discrimination of different-class features.

Our contributions include:

- designing EPSSNet, a lightweight mobile robot model, integrates semantic segmentation and boundary constraints;
- applying SAWF, a self-adaptive fusion module, which enables simultaneous learning of adaptive weights for boundary and semantic feature maps; and
- conducting comprehensive experiments on three popular datasets, demonstrating our approach's superior lightweight design and segmentation accuracy compared to existing methods, rendering EPSSNet widely applicable.

RELATED WORK

Real-Time Lightweight Semantic Segmentation

Developing lightweight architectures presents a practical approach to balancing accuracy and speed to achieve efficiency in image processing models. Lednet (Wang et al., 2019) utilized asymmetric codecs to enable real-time performance. Mobilenets (Howard et al., 2017) introduced lightweight deep neural networks using depth-separable convolutions, effectively constructing lightweight networks. TopFormer (Zhang et al., 2022) adopted tokens of multiple scales as inputs to generate scale-wise semantic features, achieving an optimal balance between precision and computational efficiency. Liteseg (Emara, 2019) employed long and short residual concatenation alongside depth-separable convolutions to strike a balance between accuracy and computational cost. FDDWNet (Liu et al., 2020) presented a lightweight model that achieves a favorable trade-off between precision and speed. MSCFNet (Gao et al., 2021) explored a structure with asymmetry in its encoder-decoder design aimed at reducing parameters without compromising accuracy. Unlike these methods, our lightweight model features two branches capable of learning semantic segmentation and boundary regions.

Boundary Processing

Adding edge computing to semantic segmentation avoids increasing computational resources (Elgendy et al., 2021) and enhances model prediction effectiveness (Lv et al., 2022). WSBE (Chen et al., 2020) utilizes CNN classifiers to locate synthetic boundary annotations for network training roughly, providing segmentation constraints. LFRSNet (Yang et al., 2022) introduces contextual and geometric features for region-awareness and segmented edges, thereby improving segmentation performance. CWAE (Marmanis et al., 2018) incorporates edge detection within the Segnet encoder-decoder architecture to enhance segmentation. Bapa-net (Liu et al., 2021) implements a prototype alignment module to improve model performance by integrating boundary and prototype alignment. BANet (Zhou et al., 2022) introduces semantic flow branching and boundary detection branching to balance accuracy and efficiency. BGCA (Ma et al., 2021) utilizes boundaries as a primary guide for contextual aggregation, facilitating overall semantic understanding. BASeg (Xiao et al., 2023) proposes using boundary information to guide context aggregation, improving performance while reducing computational effort. BPG (Wang et al., 2019) introduces a boundary-aware knowing approach, particularly in boundary regression, to monitor edge graphs, effectively guiding boundary segmentation between different regions. JSENet (Hu et al., 2020) introduces a model aimed at semantic segmentation and boundary detection, intertwining region and boundary information for enhanced segmentation. DecoupleNet (Li et al., 2020) distorts image features by learning fluency, then fuses edge features with subject features, improving segmentation accuracy while maintaining high inference efficiency. LDF (Yu et al., 2018) offers a network with distinctive features, including a boundary framework that allows the partitioning of boundary features by semantic boundary monitoring. Gated-scnn (Takikawa et al., 2019) proposes a structure with two parallel streams of CNNs, focusing on processing boundary information, effectively eliminating noise, and providing clearer predictions.

Similar to the BANet, BASeg, and Gated-scnn architectures, we propose a dual-stream approach focusing on semantic segmentation and boundary processing, respectively. However, our approach differs in the following aspects: firstly, EPSSNet is a lightweight model suitable for deployment on mobile robots (Chandani et al., 2021); secondly, the model optimizes both boundary and semantic aspects mutually. Our approach employs a self-adaptive fusion module (SAWF) to train adaptive weights for boundary and semantic feature map information, rather than simply fusing boundary and semantic features to refine the feature map. We then conduct weighted fusion of boundary and semantic feature maps. This results in a lighter model with enhanced precision compared to other models (Table 1).

Table 1. Comparative table of pros and cons of existing methods

Related work	Pros	Cons
Elgendy et al., 2021	Improve system performance and efficiency. Reduce latency. Improve user experience. Save energy consumption	Algorithm design and implementation are complex. Resource competition may lead to performance degradation. Increased network latency. Increases device energy consumption. There are security risks.
Chen et al., 2020	1. reduce dependence on labeled data. 2. Improve semantic segmentation accuracy. 3. Adapt to diverse scenarios. 4. Improve computational efficiency.	1. Boundary recognition and semantic segmentation may not be accurate for complex scenes and objects. 2. Performance may not be as good as supervised learning methods. 3. Requires significant computational resources and time
Marmanis et al., 2018	1. Improve the precision and accuracy of semantic image segmentation. 2. effectively handle complex scenes. 3. Combining classification and boundary detection methods	1. Increased computational complexity. 2. Sensitivity to noise. 3. Difficulty in parameter adjustment.
Zhou et al., 2022	1. Improve semantic segmentation accuracy and robustness. 2. A simple structure is easy to implement and train. 3. Excellent performance in handling small and irregular targets	1. Performance may not be as good as complex network structures under large-scale datasets. 2. Boundary blurring may occur when dealing with complex scenes.
Ma et al., 2021	1. Improve semantic segmentation accuracy and robustness. 2. Effectively capture the semantic information of different areas in the image. 3. Excellent performance when processing complex scenes and a large number of target images.	1. It may not be possible to capture information for complex scenes and object boundaries accurately. 2. High computational complexity. 3. There may be mis-segmentation or omission.
Xiao et al., 2023	1. Improve road boundary and obstacle recognition accuracy. 2. Improve semantic segmentation accuracy. 3. Effectively distinguish different categories of objects	1. Requires large amounts of data and computing resources. 2. There may be misidentification or missed detection in complex road scenes. 3. The model needs to be continuously updated and optimized.
Yuan et al., 2020	1. Improve semantic segmentation accuracy. 2. Reduce false segmentation. 3. Robust to occlusion and complex scenes.	1. The computational complexity is high. 2. High requirements for data annotation.

METHODS

This section introduces the comprehensive framework of the model, followed by the semantic flow branch (SFB) and the boundary flow branch (EFB). Finally, the boundary loss function and the final loss function of the semantic splitter are presented.

EPSSNet Network Architecture

The general framework of the network is depicted in Figure 3. EPSSNet comprises a Semantic Flow Branch (SFB), an Edge Flow Branch (EFB), and a Self-adapting Weighting Fusion (SAWF). A loss function supervises each branch and serves as a mutual mediator of learning roles. Since we aim to learn semantic and boundary information simultaneously, we propose a lightweight two-stream encoder to capture the corresponding features in the image. First, a simple shared stem module composed of two MobileNetV2 (Zhu et al., 2018) modules embeds the original picture $I \in \mathbb{R}^{C \times H \times W}$ into a high-dimensional feature space, where H and W represent the height and width, respectively. First, a simple shared stem module composed of two MobileNetV2 (Zhu et al., 2018) modules embeds the original picture $I \in \mathbb{R}^{C \times H \times W}$ into a high-dimensional feature space, where H and W represent the height and width, respectively. The semantic flow branch's height and width generate semantically rich features. We emphasize that the semantic flow can be any lightweight semantic segmentation backbone, such as SegFormer (Xie et al., 2021), TopFormer (Zhang et al., 2022), or liwFormer (Dong et al., 2023). In this paper, we choose one of the latest SOTA methods, AFFormer-T (Bo et al., 2023) ("T" stands for tiny model of AFFormer) as our semantic flow backbone.

At the end of the backbone, the ASPP module (Chen et al., 2017) compresses the number of channels from 2048 to 512 to enhance spatial detail extraction. The final feature map produced by the SFB is denoted as F_s.

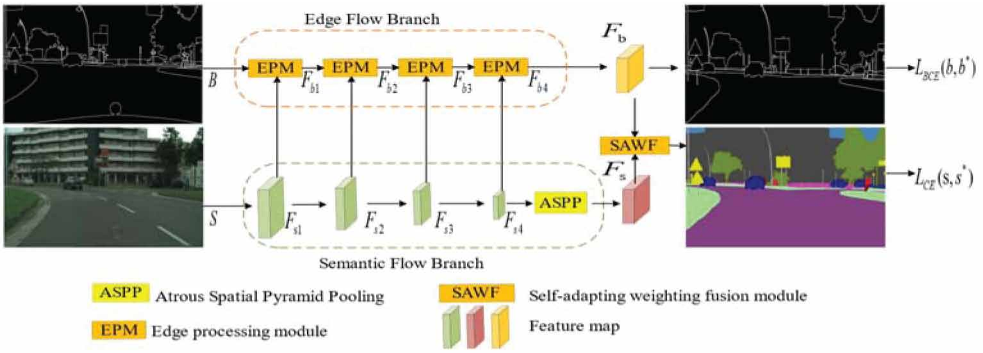
In EFB, we input the boundary features into the Edge processing module (EPM), which later fuses the feature maps generated by multiple EPMs and the backbone network so that the boundary features are augmented while the non-object boundaries are suppressed. EPB highlights the boundary information by aggregating the multi-level semantic information, and the resultant feature map of the boundary information is $\boldsymbol{F_b}$. For the i-th position, the information conversion process in the boundary flow is expressed as: $f = Q \otimes (\sigma(\Gamma(Cv_2(\psi(Cv_1(\mathrm{Concat}(Q, F_{si}))))))) + F_{si}$, where Q represents the boundary feature of the i-th stage, Cv_1 and Cv_2 stand for 1×1 convolutions, Ψ stands for global average pooling, Γ stands for Relu activation function, and σ stands for Sigmoid function, respectively. Then, the features of different fields are effectively integrated through the self-adapting weighing fusing module. To generate the predictive feature maps, we input $\boldsymbol{F_s}$, $\boldsymbol{F_b}$ to the SAWF module to fuse the features of the two branches adaptively.

Edge Flow Brance

In semantic segmentation tasks, inconsistent boundary segmentation often arises, especially in large areas and complex scenes, primarily due to the absence of context. In semantic segmentation tasks, inconsistent boundary segmentation often appears, especially in large areas and complex scenes, primarily due to the absence of context. To address this, the incorporation of global context through average pooling is employed. However, while global context offers high semantic information, it lacks spatial detail. Therefore, we need various receptive views to refine spatial information.

Nevertheless, different scales of receptive views may produce varying discriminatory features, leading to inconsistency. Thus, selecting more discriminative features is essential for predicting unified boundary labels. Our network comprises four stages based on feature map size, each exhibiting varying recognition capabilities and consistent performance. The Edge Flow Branch (EFB) encodes refined boundary information but lacks spatial context guidance, resulting in poorer boundary consistency. Conversely, the Semantic Flow Branch (SFB) provides strong semantic consistency but coarse spatial

Figure 3. Detailed architecture of the overall framework of the EPSSNet model



prediction. To leverage their strengths, we propose EPSSNet, a lightweight model combining boundary and semantic features for optimal prediction.

Additionally, as shown in Figure 4, to enhance boundary consistency, we introduce the Edge Processing Module (EPM), which computes channel attention vectors by combining boundary and semantic features from adjacent stages. The EPM adjusts feature weights to enhance boundary information consistency across stages. The calculation formula of EPM is shown in Formula 1:

$$f = Q \otimes (\sigma(\Gamma(Cv_2(\psi(Cv_1(Concat(Q,K))))))) + K \tag{1}$$

Here, Cv_1, Cv_2 stand for 1×1 convolutions, ψ stands for global average pooling, Γ stands for Relu activation function, and σ stands for Sigmoid function, respectively, Q stands for boundary feature map, K stands for semantic feature map.

Formula (1) demonstrates the variation in discriminative capabilities across different stages of features. To make the boundary information of each stage consistent, we need the Sigmoid function to identify the input features. Through a design such as EFB, the network can obtain judgmental features in stages, thereby making the boundary information consistent.

Self-Adapting Weighting Fusion

After acquiring high-level boundary features and semantic features from the boundary flow branch, the next step is effectively merging these two sets of features. Since semantic feature maps result from supervised learning of semantic segmentation tasks and boundary features are obtained through supervised learning of boundary losses, a significant disparity exists between these two types of features. Many current approaches, such as TopFormer and Isdbes, employ fixed-weight methods where learned weights are independent of input features. However, the importance of boundary and semantic features may vary across different images, suggesting that weight parameters should be closely related to input features. Addressing these concerns, this paper proposes a method that integrates relationship attention, boundary feature attention, and semantic feature attention, facilitating the fusion of boundary and semantic features, as depicted in Figure 5.

Given the boundary feature map $F_b \in \mathbb{R}^{C \times W \times H}$, and semantic feature map $F_s \in \mathbb{R}^{C \times W \times H}$, here, C, H, and W denote the number of channels, height, and width of the feature map, respectively. We designed a SAWF module that can learn the spatial attention map. The map size is $H_{sb}^{*} \times W_{sb}^{*}$. We utilize the C-dimensional feature vector at each spatial position as the feature representation. All spatial locations form an attention graph consisting of N = $H_{sb}^{*} \times W_{sb}^{*}$ nodes. As shown in Figure 5, we assign spatial location identification numbers as 1, …., N. We denote N eigenvectors as $x_i \in \mathbb{R}^{C}$, where i = 1, …, N.

Figure 4. Framework diagram of EPM

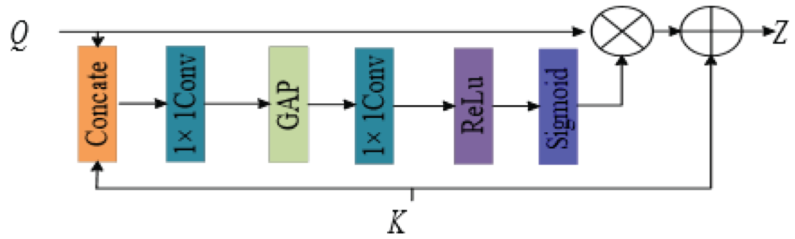
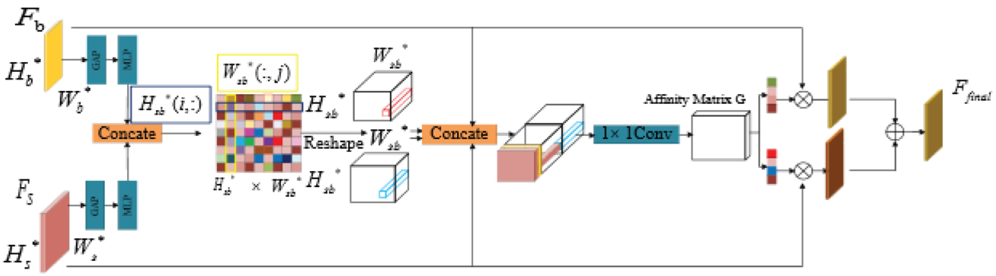


Figure 5. Detailed structure of the network module of SAWF



The relationship between feature vector i and feature vector j is represented by $r_{i,j}$. $r_{i,j}$ can be described as the dot product affinity between feature vectors. The calculation Formula (2) of $r_{i,j}$ is as follows:

$$r_{i,j} = f_s\left(x_i, x_j\right) = M_{s1}(G_{s1}\left(F_b\right))^T M_{s2}(G_{s2}\left(F_s\right)) \tag{2}$$

Among them, M_{s1}, M_{s2}, G_{s1}, G_{s2}, represent shared multi-layer perceptron and global average pooling, respectively, $r_{i,j} \in \mathbb{R}^{H_{sb}^* \times W_{sb}^*}$. $H_{sb}^*\left(i,:\right)$ represents the affinity relationship between the i-th row and other rows. $H_{sb}^*\left(:,j\right)$ represents the affinity relationship between the jth column and other columns. After that, the feature map is reshaped according to the columns and rows of the affinity attention map, and the reshaped feature map is stacked alongside the semantic feature map and the boundary feature map, ensuring the robustness of the stacked feature map. The stacked feature map calculation process is shown in Formula 3:

$$y_{sb} = concat(reshape(H^*{}_{sb}(i,:), H^*{}_{sb}(:,j))) \tag{3}$$

Here, $y_{sb} \in \mathbb{R}^{C \times W \times H}$. Semantic and boundary features are generated under the supervision of different loss functions, so we first associate the two types of information and then use dynamic weights to fuse boundary and semantic information effectively. Given $y_{sb} \in \mathbb{R}^{C \times W \times H}$, after convolution, we get the semantic feature vector $F_s \in \mathbb{R}^{C \times 1 \times 1}$ and the boundary feature vector $F_b \in \mathbb{R}^{C \times 1 \times 1}$. Then, the two feature vectors are stacked according to the channel direction, and then the stacked vectors are divided into H groups. Finally, the affinity matrix is calculated, generating the semantic feature weight and boundary feature weight from this matrix. The specific calculation formula is as Formula 4:

$$\begin{aligned} Q_s &= Cv_{3\times3}\left(F_s\right), W_b = Cv_{3\times3}\left(F_b\right) \\ E_{sb} &= Concat(Q_s, F_b) \\ R_{i,j} &= \frac{e^{q_i k_j}}{\sum_{i=1}^{H} e^{q_i}} \end{aligned} \tag{4}$$

Here, q_i denotes the query vector of the i -th head of multi-head attention and the key vector of the j -th head. Then, the adaptive fusion weight of the i -th head is shown in Formula 5:

$$w_i = Rv_i \tag{5}$$

Here, q_i represents the v vector of the i -th head. We divide the weight vector w_i into a semantic feature vector w_s and a boundary feature vector w_b according to Formula 3, then the feature calculation process after the fusion of semantic features and boundary features is as shown in Formula 6:

$$O_f = \left(1 + w_s\right) Q_s + \left(1 + w_b\right) F_b \tag{6}$$

Here, $Q_f \in \mathbb{R}^{C \times W \times H}$ is the feature map that fuses the semantic feature map and the boundary feature map, where C denotes the number of categories. Overall, SAWF can estimate the relationships within and between semantic features and boundary features to learn optimal fusion weights.

Loss Function

Throughout the training procedure, we utilized three distinct loss functions: semantic loss, boundary loss, and segmentation loss. These functions facilitated the optimization of both branches. First, the semantic features produced by the semantic stream, we use the semantic loss function, i.e., the standard multi-class cross-entropy loss $L_{BCE}\left(b, b^*\right)$, and the multi-class cross-entropy loss $L_{BCE}\left(b, b^*\right)$ is shown in Formula (7):

$$L_{CE}(s, s^*) = -\frac{1}{N} \sum_i \sum_{c=1}^{M} y_{ic} \log(p_{ic}) \tag{7}$$

Here, M represents the number of categories and y_{ic} represents the sign function, which takes the value of zero or one. If the number of truth categories of sample i is equal to c, then one is taken. Otherwise, it takes 0. p_{ic} denotes the predicted probability of observing the sample i belonging to the category c.

A binary cross-entropy loss function is used for the border flow branches to optimize the similarity between the predicted and real borders. The function is shown in Formula (8):

$$L_{BCE}(b, b^*) = -y_i \log(p_i) - (1 - y_i) \log(1 - p_i) \tag{8}$$

Here, y_i represents the label of sample i, where it equals 1 for positive class samples and 0 otherwise. p_i denotes the probability assigned to sample i being predicted as a positive class. For samples predicted as positive class, a higher probability corresponds to a smaller loss value.

Finally, to assist network training, we add pixel-level semantic segmentation cross-entropy loss L_{ACE} at the end of the backbone network. As shown in Figure 2 and Figure 4, the overall loss function is shown in Formula (9):

$$Loss = \lambda_1 L_{ACE} + \lambda_2 L_{CE}(s, s^*) + \lambda_3 L_{BCE}(b, b^*) \tag{9}$$

In order to effectively combine the three loss optimization models, we set λ_1 to 0.4, λ_2 to 1, and λ_3 to 0.8 respectively.

EXPERIMENTS

Datasets and Evaluation Metrics

To assess the validity of our approach, we extensively evaluated it on three commonly utilized datasets. These are:

- Cityscapes (Cordts et al., 2016): This dataset focuses on urban streetscape segmentation, featuring 30 categories, though only 19 were evaluated. The dataset consists of 5000 meticulously labeled images, partitioned into 2975 for training, 500 for validation, and 1525 for testing purposes.
- PASCAL VOC 2012 (Everingham & Winn, 2012): This dataset builds upon VOC2007, containing 20 categories (excluding background) with 2,913 labeled images for segmentation. The training set comprises 1,464 images, containing a collective count of 3,507 objects. Conversely, the validation set consists of 1,449 images featuring a total of 3,422 objects.
- ADE20K (Zhou et al., 2017): This dataset covers landscapes, objects, partial elements, and subcomponents, comprising 25,000 images depicting various natural scenes. Each image, on average, encompasses 19.5 instances spanning 10.5 object classes. The dataset consists of 20,210 training images, 2,000 validation images, and 3,000 testing images.

For evaluation, we used the following quantitative metrics:

Intersection-over-Union (IOU) is computed for each category, serving as a widely adopted metric. Mean Intersection-over-Union (mIoU) is utilized to gauge segmentation accuracy. F-score for edge detection, where thresholds control bias (Marmanis et al., 2018). Boundary IoU (BIoU) for semantic and binary boundary performance (Cheng et al., 2021) is more sensitive to minor object errors. We also assessed the model's FLOPs, number of parameters, and FPS on RTX 3060 GPUs. The commonly used calculation processes of IoU and mIoU are shown in Formulas 10 and 11:

$$IOU = \frac{p_{ti}}{\sum_{j=0}^{k} p_{tj} + \sum_{j=0}^{k} p_{\mu} - p_{ti}} i = 0,1,2 \ldots k, \tag{10}$$

$$\mathrm{mIoU} = \frac{1}{K+1} \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}} \tag{11}$$

Implementation Details

The Stochastic Gradient Descent (SGD) algorithm (Krizhevsky et al., 2012), with a momentum of 0.9 and weight decay of 0, is employed. The initial learning rate is $\left(1 - \frac{iter}{max\_iter}\right)^{power}$ in training and validation on the three datasets. Throughout the training and validation processes across all three datasets, the following hyperparameters were utilized: a learning rate of 0.0025, value decay of 0.9, and momentum of 0.0005. Additionally, custom image cropping dimensions were applied during training and validation: 1024 x 512 for Cityscape, 512 x 512 for PASCAL VOC 2012, and 520 x 520 for ADE20k. In order to augment the dataset, input images underwent random scaling between 0.5

and 2, as well as horizontal flipping, throughout the training process. For Cityscape, the batch size was 4, while for PASCAL VOC 2012 and ADE20K, it was 8. The training comprised 160k iterations for Cityscape, 100k iterations for PASCAL VOC 2012, and 200k iterations for ADE20K.

Ablation Study

To exhibit the authenticity of semantic flow branching and boundary flow branching, we conducted ablation trials to demonstrate that dual-branch learning surpasses the conventional single-branch semantic segmentation task, as depicted in Table 2. We chose SeaFormer as the Backbone, and we deleted the ASPP module in EPB, SAWF, and SFB in EPSSNet at the same time. The FLOPs(G) and mIoU(%) obtained in EPSSNet are 8.0 and 76.1, and the FLOPs(G) and mIoU(%) obtained in the subsequent experiments by adding SFB, EPB, and SAWF sequentially are 8.1 and 77.2, 8.5 and 78.1, and 8.7 and 79.6, respectively. The better the experimental results obtained.

Finally, we utilize the Boundary Class Intersection Ratio (BIoU) to evaluate the accuracy of semantic and boundary flow branching. As illustrated in Figure 6, the values correspond to 3, 5, 9, and 12 pixels when using different thresholds of 0.0003, 0.0005, 0.0009, and 0.00012, respectively.

Furthermore, we incorporated the boundary flow F-Score into the Seaformer backbone network, resulting in a more precise and more accurate outcome than the boundary features learned by using Seaformer alone. Different thresholds can yield varied scores, highlighting the importance of aligning boundary predictions with actual boundaries by adjusting the thresholds to minimize bias.

To prove the effectiveness of the model in boundary feature extraction and processing, we visualize the boundary feature maps of each stage of the Edge Flow Branch, including, F_{b1}, F_{b2}, F_{b3} and F_{b4}. As depicted in Figure 7, the predicted image becomes more defined as the network increases

Table 2. Ablation experiments on the cityscapes dataset

Baseline	SFB	EPB	SAWF	FlOPs(G)	mIoU(%)
✓				8.0	76.1
✓	✓			8.1	77.2
✓	✓	✓		8.5	78.1
✓	✓	✓	✓	8.7	79.6

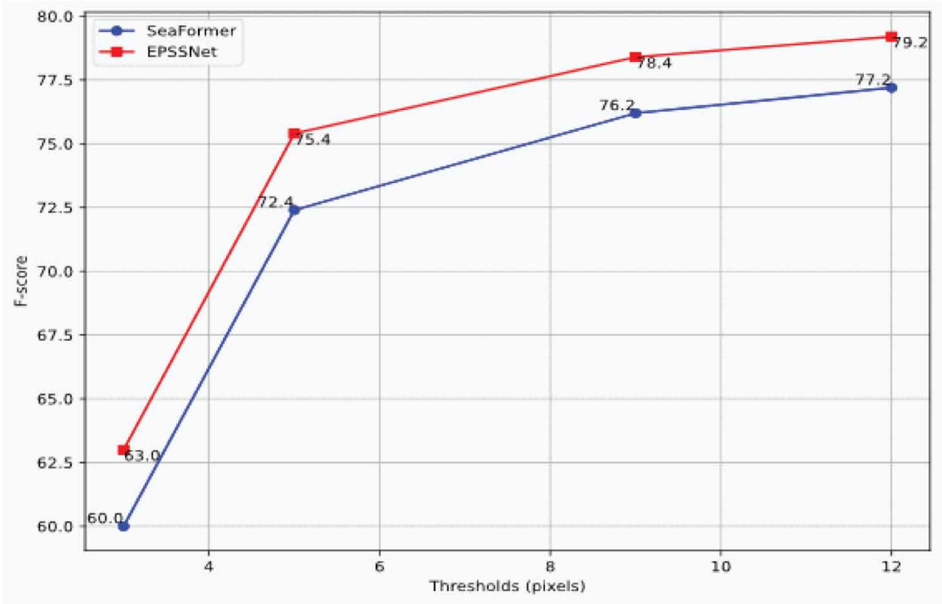
Table 3. Ablation experiments on the cityscapes dataset

Cv_1	Cv_2	GAP	FPS	mIoU(%)
✓			22	79.3
✓	✓		20	79.8
✓	✓	✓	23	79.6

Table 4. Ablation experiments on SAWF parts on the cityscapes dataset

AM_1	Concat_1	AM_2	Concat_2	mIoU(%)
✓	✓			78.5
✓	✓	✓		78.6
✓	✓	✓	✓	79.6

Figure 6. BIoU metrics for EPSSNet vs. seaformer on the cityscapes dataset



in depth, and the adapted image aligns more closely with the original image. This also indicates that employing successive EPM modules aids in the recovery of boundary details.

Comparisons With State-of-the-Art Methods

To demonstrate the efficiency and applicability of our method, we performed comparative analyses on three datasets: Cityscapes, PASCAL VOC 2012, and ADE20k. Our model attained a mIoU score of 80.2%, representing a 2.7% enhancement compared to earlier models like Light Head(f) and LR-ASPP, as detailed in Table 5, ensuring equitable comparison. Figure 8 illustrates the visualization results of EPSSNet alongside other classical lightweight models. Notably, EPSSNet exhibits enhanced delineation of target shapes, resulting in smoother and finer boundaries. The illustrated instances comprise a lamppost in the initial row, a footpath in the subsequent row, and a bike in the fourth row.

Figure 8 illustrates the boundary detection performance of different lightweight models on the Cityscape validation dataset. Our method, EPSSNet, demonstrates its effectiveness in accurately capturing boundaries and minimizing extraneous noise. Specifically, it adeptly represents road edges,

Figure 7. Visualization of Edge prediction maps at different stages flow (Note: (a) Image; (b) Lab; (c) F_{b1} stage edge feature ma; (d) F_{b2} stage edge feature map; (e) F_{b3} stage edge feature ma; (f) F_{b4} stage edge feature map)

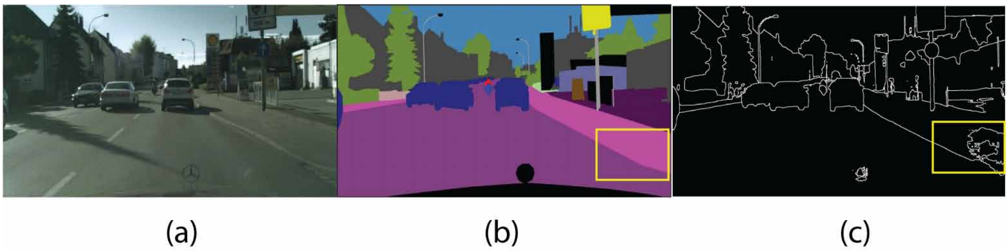


Table 5. Results on the cityscapes validation dataset and test dataset—represents some methods without test results

Method	Backbone	FLOPs(G)	mIoU(val)	mIoU(test)
FCN	MobileNetV2 (Sandler et al., 2018)	317	61.5	-
PSPNet	MobileNetV2 (Sandler et al., 2018)	423	70.2	-
SegFormer(f)	MiT-B0 (Xie et al., 2021)	125.5	76.2	-
L-ASPP	MobileNetV2 (Sandler et al., 2018)	12.6	72.7	-
LR-ASPP	MobileNetV3-L (Howard et al., 2019)	9.7	72.4	72.6
LR-ASPP	MobileNetV3-S (Howard et al., 2019)	2.9	68.4	69.4
Simple Head(h)	TopFormer-B (Zhang et al., 2022)	2.7	70.7	-
Simple Head(f)	TopFormer-B (Zhang et al., 2022)	11.2	75.0	75.0
Light Head(f)	SeaFormer-B (Wan et al., 2023)	13.7	77.7	77.5
EPSSNet	SeaFormer-B (Wan et al., 2023)	15.6	79.6	80.2

lawns with diverse elevations, and various object shapes, as highlighted in the yellow box and across different rows. This underscores the notable improvement in model performance attributable to our proposed boundary flow block.

To showcase the overall effectiveness of our method, we visually contrasted our model with its less complex alternative on the PASCAL VOC 2012 validation dataset, as illustrated in Figure 9. The figure indicates that our method performs well even in small regions, such as the first line of an airplane tail. The other approaches fail to delineate the entirety of our segmentation results due to the initial two, the subsequent line of mutton.

These alternative methods are susceptible to background noise interference and are incapable of consistent segmentation, whereas our method adeptly suppresses noise and achieves uninterrupted segmentation of miniature objects. The third row's horse leg and the fourth row's bicycle are examples of inconsistency within the class. This is the same reason as the sheep's leg in the second row. However, our segmentation effect shows a regionally significant improvement compared to the former, which results in intra-class consistency and completeness. This demonstrates our integration of multi-level semantic characteristics and boundary features, resulting in improved feature maps and refined boundary details.

To highlight the efficiency of our model, we conducted comparisons between EPSSNet and other established models in terms of parameter count, FLOPs, FPS, and mIoU. FLOPs (Floating Point Operations Per Second) are a metric for evaluating the efficiency of computer programs, algorithms, or hardware. It quantifies the number of floating-point operations conducted within a specific duration. FPS (Frames Per Second) is a measure employed to assess the efficiency of an image or video processing system, denoting the rate at which frames are handled within a given time frame. mIoU (mean Intersection over Union) is a widely utilized assessment criterion in semantic segmentation endeavors. It gauges the precision of the model's segmentation outcomes at the pixel level.

The comparative results are summarized in Table 6. It is evident that our method shows a diminished number of parameters, lowered computational burden, and superior mIoU achievement compared to other methodologies. This confirms our achievement in striking a favorable balance between accuracy and efficiency by integrating semantic flow and boundary flow branches.

To further highlight the advantages of our approach, we conducted a quantitative comparison in accuracy (mIoU) and inference speed (FPS) against other existing lightweight models, as depicted in Figure 10. Our approach attained an accuracy rate of 80.2% with a processing speed of 23 frames per second, marginally behind DRNet-39 in accuracy and lagging in speed compared to STD-Seg75. However, it still outperforms them in terms of the combined metric of speed and accuracy. These

Figure 8. Boundary detection performance

(Note: On the Cityscapes validation dataset, our method's visualization results are juxtaposed with those of other methods, with images displayed from left to right, followed by corresponding labels. The sequence includes SegFormer, SeaFormer, and EPSSNet.)

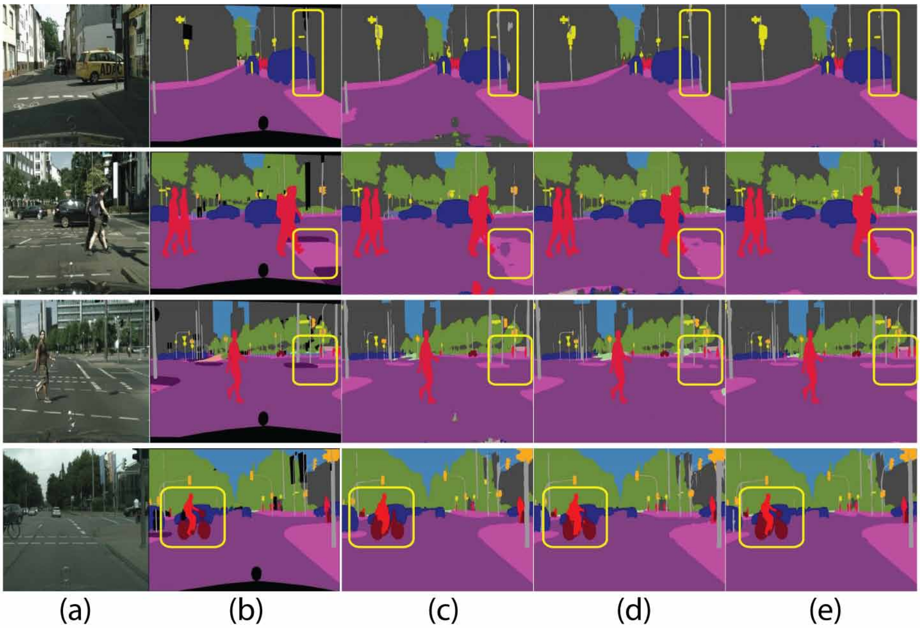


Figure 9. PASCAL VOC 2012 validation dataset

(Note: For the Cityscapes validation dataset, the comparison of boundary detection results between our method and others is presented. Images are displayed from left to right, followed by their corresponding labels. The sequence includes SegFormer, SeaFormer, and EPSSNet.)

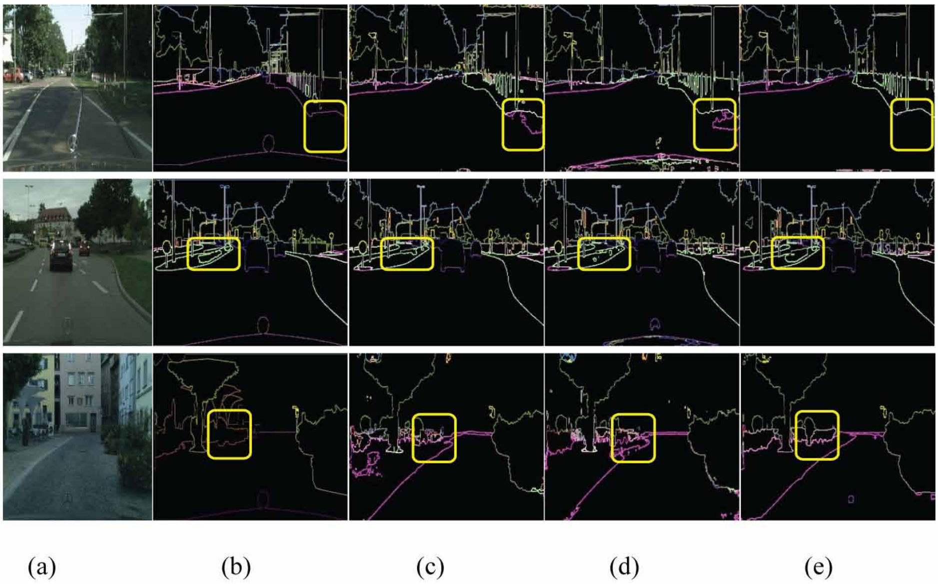
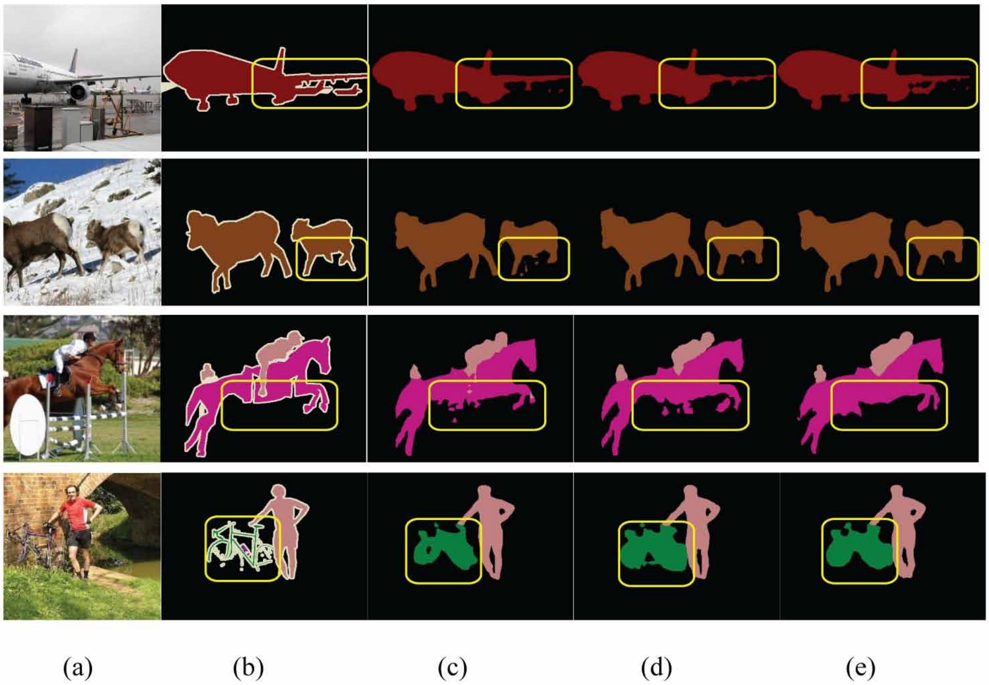


Table 6. Semantic segmentation results on cityscapes val dataset

Method	#Params	FLOPs(G)	MIoU(%)	FPS
FCN (Long et al., 2015) [12]	9.8M	317G	61.5	11.2
PSPNet (Zhao et al., 2017) [6]	13.7M	423G	70.2	9.5
DeepLab V3+ (Chen et al., 2018) [13]	15.4M	555G	75.2	8.2
Lednet (Wang et al., 2019)	13.2M	354G	70.1	9.1
FDDWNet (Liu et al., 2020)	16.1M	421G	65.3	8.6
JSENet (Hu et al., 2020)	16.3M	462G	64.5	8.4
WSBE (Chen et al., 2020)	14.2M	445G	69.1	7.2
MSCFNet (Gao et al., 2021)	15.1M	578G	70.3	6.5
Bapa-net (Liu et al., 2021)	7.5M	117G	75.2	11.1
LSRSNet(Yang et al., 2022)	8.2M	114.3G	76.1	10.2
SegFormer-B0 (Xie et al., 2021)[14]	3.8M	125G	76.2	11.7
TopFormer-B (Zhang et al., 2022)[15]	5.1M	11.2G	75.2	55.6
PIDNet-S(Xu et al., 2023) [16]	7.6M	47.6G	78.7	15.3
LRFormer-T* (Wu et al., 2023) [17]	13.0M	122.0G	80.7	-
AFFormer-B (Bo et al., 2023)[18]	3.0M	33.5G	77.8	21.2
Ours	2.4M	31.5G	80.9	22.6

Figure 10. Visualization Comparison on the PASCAL VOC 2012 validation dataset (Note: From left to right: (a) Image, (b) Label, (c) SegFormer, (d) SeaFormer, (e) EPSSNet)



findings highlight the method's capacity to achieve a well-balanced compromise between precision and efficiency.

In addition, in order to prove that our model can be lightweight while maintaining high accuracy, we compared it with existing methods on the Cityscapes dataset. As shown in Figure 1, although our method is slightly less accurate than DDRNet-39 (Hong et al., 2021), it is faster in terms of speed. Compared with other methods, our segmentation index is higher.

Finally, the experimental outcomes validate the practical applicability of our approach through quantitative and qualitative assessments across diverse datasets. EPSSNet showcases its capacity to strike a commendable equilibrium between precision and speed, rendering it suitable for deployment in lightweight robot mobility platforms for tasks like automated robot segmentation and detection.

CONCLUSION

This paper introduces the EPSSNet model, a lightweight autonomous edge processing and semantic segmentation network for mobile robots. EPSSNet consists of semantic flow branches (SFB) and boundary flow blocks (EFB). SFB is used to attract the same pixels and repel different pixels, thereby achieving precise segmentation of objects and content, while EFB generates semantic information through the internal consistency of objects to detect boundary information and guide the formation of boundary areas. In addition, the SAWF module is able to learn weights and adaptively fuse boundary features and semantic features. Evaluation results show that the EPSSNet model outperforms existing models on three datasets, confirming its feasibility in various applications such as semantic SLAM, semantic perception in the environment, and robot detection and control. To augment the effectiveness and adaptability of the EPSSNet model and expand its utility, forthcoming endeavors might concentrate on the subsequent domains: (a) integrating additional depth information with EPSSNet's visual data to enhance semantic segmentation and boundary detection performance by improving environmental understanding; (b) enhancing the real-time performance of EPSSNet through optimization techniques such as model compression, lightweight network design, and hardware acceleration; (c) extending the application of EPSSNet to various robot tasks, including environmental monitoring, surveillance, human-computer interaction, and intelligent navigation, thereby expanding its utility across diverse domains; and (d) tailoring the EPSSNet model to the specific characteristics of different robot platforms to ensure optimal performance on varying hardware setups.

COMPETING INTERESTS

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

FUNDING

No funding was received for this work.

REFERENCES

Barbosa, A., Bittencourt, I. I., Siqueira, S., Dermeval, D., & Cruz, N. J. (2022). A context-independent ontological linked data alignment approach to instance matching. [IJSWIS]. *International Journal on Semantic Web and Information Systems*, *18*(1), 1–29. doi:10.4018/IJSWIS.295977

Bo, D., Pichao, W., & Wang, F. (2023). Afformer: Head-free lightweight semantic segmentation with linear transformer. *Proceedings of the AAAI Conference on Artificial Intelligence*, *37*(1), 516–524. doi:10.1609/aaai.v37i1.25126

Chandani, A., Sriharshitha, S., Bhatia, A., Atiq, R., & Mehta, M. (2021). Robo-advisory services in India: A study to analyse awareness and perception of millennials. [IJCAC]. *International Journal of Cloud Applications and Computing*, *11*(4), 152–173. doi:10.4018/IJCAC.2021100109

Chen, L., Papandreou, G., Kokkinos, I., Murphy, K. P., & Yuille, A. L. (2017). DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(4), 834–848. doi:10.1109/TPAMI.2017.2699184 PMID:28463186

Chen, L., Wu, W., Fu, C., Han, X., & Zhang, Y. (2020). Weakly supervised semantic segmentation with boundary exploration. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), Lecture notes in computer science: Vol. 12371. *Computer Vision—ECCV 2020* (pp. 347–362). Springer., doi:10.1007/978-3-030-58574-7_21

Cheng, B., Girshick, R. B., Doll'ar, P., Berg, A. C., & Kirillov, A. (2021). Boundary IoU: Improving object-centric image segmentation evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp.15329–15337). IEEE. doi:10.1109/CVPR46437.2021.01508

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., & Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3213–3223). IEEE. doi:10.1109/CVPR.2016.350

Deveci, M., Pamučar, D., Gokasar, I., Köppen, M., & Gupta, B. B. (2022). Personal mobility in metaverse with autonomous vehicles using Q-rung orthopair fuzzy sets based OPA-RAFSI model. *IEEE Transactions on Intelligent Transportation Systems*, *24*(12), 15642–1565. doi:10.1109/TITS.2022.3186294

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021, May 3–7). An image is worth 16x16 words: Transformers for image recognition at scale [Conference presentation]. *International Conference on Learning Representations*. Open Review. https://openreview.net/forum?id=YicbFdNTTy

Elgendy, I. A., Zhang, W., He, H., Gupta, B. B., & Abd El-Latif, A. A. (2021). Joint computation offloading and task caching for multi-user and multi-task MEC systems: Reinforcement learning-based algorithms. *Wireless Networks*, *27*(3), 2023–2038. doi:10.1007/s11276-021-02554-w

Emara, T., Munim, H. A., & Abbas, H. M. (2019). Liteseg: A novel lightweight convnet for semantic segmentation. In 2019 Digital Image Computing: Techniques and Applications (DICTA) (pp.1–7). IEEE. doi:10.1109/DICTA47822.2019.8945975

Everingham, M., & Winn, J. (2012). The PASCAL visual object classes challenge 2012 (VOC2012) development kit. *Pattern Analysis, Statistical Model and Computational Learning Technical Report, 2007*(1-45), 5.

Gao, G., Xu, G., Yu, Y., Xie, J., Yang, J., & Yue, D. (2021). MSCFNet: A lightweight network with multi-scale context fusion for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, *23*(12), 25489–25499. doi:10.1109/TITS.2021.3098355

Gu, J., Li, G., Vo, N. D., & Jung, J. J. (2022). Contextual Word2Vec model for understanding chinese out of vocabularies on online social media. [IJSWIS]. *International Journal on Semantic Web and Information Systems*, *18*(1), 1–14. doi:10.4018/IJSWIS.309428

Howard, A. G., Sandler, M., Chu, G., Chen, L., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q. V., & Adam, H. (2019). Searching for MobileNetV3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 1314–1324). IEEE. doi:10.1109/ICCV.2019.00140

Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 7132–7141). IEEE. doi:10.1109/CVPR.2018.00745

Hu, Z., Zhen, M., Bai, X., Fu, H., & Tai, C. (2020). JSENet: Joint semantic segmentation and edge detection network for 3D point clouds. In A. Vedaldi, H. Bischof, T. Brox, & J. M. Frahm (Eds.), Lecture notes in computer science: Vol. 12365. *Computer Vision – ECCV 2020* (pp. 222–239). Springer. doi:10.1007/978-3-030-58565-5_14

Ismail, S., Shishtawy, T. E., & Alsammak, A. K. (2022). A new alignment word-space approach for measuring semantic similarity for Arabic text. [IJSWIS]. *International Journal on Semantic Web and Information Systems*, *18*(1), 1–18. doi:10.4018/IJSWIS.297036

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90. doi:10.1145/3065386

Li, X., Li, X., Zhang, L., Cheng, G., Shi, J., Lin, Z., Tan, S., & Tong, Y. (2020). *Improving semantic segmentation via decoupled body and edge supervision.* Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK. doi:10.1007/978-3-030-58520-4_26

Li, X., Wang, W., Hu, X., & Yang, J. (2019). Selective kernel networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 510–519). IEEE. doi:10.1109/CVPR.2019.00060

Liu, J., Zhou, Q., Qiang, Y., Kang, B., Wu, X., & Zheng, B. (2020). FDDWNet: A lightweight convolutional neural network for real-time semantic segmentation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 2373–2377). IEEE. doi:10.1109/ICASSP40776.2020.9053838

Liu, R. W., Guo, Y., Lu, Y., Chui, K. T., & Gupta, B. B. (2022). Deep network-enabled haze visibility enhancement for visual IoT-driven intelligent transportation systems. *IEEE Transactions on Industrial Informatics*, *19*(2), 1581–1591. doi:10.1109/TII.2022.3170594

Liu, Y., Cheng, M., Bian, J., Zhang, L., Jiang, P., & Cao, Y. (2022). Semantic edge detection with diverse deep supervision. *International Journal of Computer Vision*, *130*(1), 179–198. doi:10.1007/s11263-021-01539-8

Liu, Y., Cheng, M., Hu, X., Wang, K., & Bai, X. (2017). Richer convolutional features for edge detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition* (pp. 3000–3009). IEEE.

Liu, Y., Deng, J., Gao, X., Li, W., & Duan, L. (2021). Bapa-net: Boundary adaptation and prototype alignment for cross-domain semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 8781–8791). IEEE. doi:10.1109/ICCV48922.2021.00868

Liu, Y., Zhou, Q., Wang, J., Wang, Z., Wang, F., Wang, J., & Zhang, W. (2023). Dynamic token-pass transformers for semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 1827–1836). IEEE. doi:10.48550/arXiv.2308.01944

Liu, Z., Chen, L., Tong, L., Zhou, F., Jiang, Z., Zhang, Q., Shan, C., Wang, Y., Zhang, X., Li, L., & Zhou, H. (2022). Deep learning based brain tumor segmentation: A survey. *Complex & Intelligent Systems*, *9*(1), 1001–1026. doi:10.1007/s40747-022-00815-5

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3431–3440). IEEE. doi:10.1109/CVPR.2015.7298965

Luo, W., Li, Y., Urtasun, R., & Zemel, R. (2016). Understanding the effective receptive field in deep convolutional neural networks. In *30th Conference on Neural Information Processing Systems.* IEEE. doi:10.48550/arXiv.1701.04128

Lv, L., Wu, Z., Zhang, L., Gupta, B. B., & Tian, Z. Q. (2022). An edge-AI based forecasting approach for improving smart microgrid efficiency. *IEEE Transactions on Industrial Informatics*, *18*(11), 7946–7954. doi:10.1109/TII.2022.3163137

Marmanis, D., Schindler, K., Wegner, J. D., Galliani, S., Datcu, M., & Stilla, U. (2018). Classification with an edge: Improving semantic image segmentation with boundary detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, *135*, 158–172. doi:10.1016/j.isprsjprs.2017.11.009

Mohammed, S. S., Menaouer, B., Zohra, A. F. F., & Nada, M. (2022). Sentiment analysis of COVID-19 tweets using adaptive neuro-fuzzy inference system models. [IJSSCI]. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–20. doi:10.4018/IJSSCI.300361

Nguyen, G. N., Viet, N. H., Elhoseny, M., Shankar, K., Gupta, B. B., & El-Latif, A. A. (2021). Secure blockchain enabled Cyber–physical systems in healthcare using deep belief network with ResNet model. *Journal of Parallel and Distributed Computing*, *153*, 150–160. doi:10.1016/j.jpdc.2021.03.011

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., & Lerer, A. (2017, December 9). *Automatic differentiation in PsyTorch* [Conference presentation]. NIPS 2017 Autodiff Workshop, Long Beach, CA, USA. https://openreview.net/pdf?id=BJJsrmfCZ

Salhi, D. E., Tari, A., & Kechadi, M. T. (2021). Using e-reputation for sentiment analysis: Twitter as a case study. [IJCAC]. *International Journal of Cloud Applications and Computing*, *11*(2), 32–47. doi:10.4018/IJCAC.2021040103

Sandler, M., Howard, A. G., Zhu, M., Zhmoginov, A., & Chen, L. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4510-4520). IEEE. doi:10.1109/CVPR.2018.00474

Sharma, K., Anand, D., Mishra, K. K., & Harit, S. (2022). Progressive study and investigation of machine learning techniques to enhance the efficiency and effectiveness of industry 4.0. [IJSSCI]. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–14. doi:10.4018/IJSSCI.300365

Singh, S. K., & Sachan, M. K. (2021). Classification of code-mixed bilingual phonetic text using sentiment analysis. [IJSWIS]. *International Journal on Semantic Web and Information Systems*, *17*(2), 59–78. doi:10.4018/IJSWIS.2021040104

Sun, Z., Wu, B., Xu, C., & Kong, H. (2019). Gated-SCNN: Gated shape CNNs for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 5228–5237). IEEE. doi:10.1109/ICCV.2019.00533

Sun, Z., Wu, B., Xu, C., & Kong, H. (2022). Ada-detector: Adaptive frontier detector for rapid exploration. In *2022 International Conference on Robotics and Automation (ICRA)* (pp.3706–3712). IEEE. doi:10.1109/ICRA46639.2022.9811614

Takikawa, T., Acuna, D., Jampani, V., & Fidler, S. (2019). *Gated-scnn: Gated shape cnns for semantic segmentation*. ICCV.

Talib, L. F., Amin, J., Sharif, M., & Raza, M. (2024). Transformer-based semantic segmentation and CNN network for detection of histopathological lung cancer. *Biomedical Signal Processing and Control*, *92*, 106106. doi:10.1016/j.bspc.2024.106106

Teichmann, M., Weber, M., Zöllner, J. M., Cipolla, R., & Urtasun, R. (2018). MultiNet: Real-time joint semantic reasoning for autonomous driving. In *2018 IEEE Intelligent Vehicles Symposium (IV)* (pp. 1013–1020). IEEE. doi:10.1109/IVS.2018.8500504

Vaswani, A., Shazeer, N. M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *31st Conference on Neural Information Processing Systems(NIPS2017). arXiv.* doi:10.48550/arXiv.1706.03762

Wang, B., Qi, G., Tang, S., Zhang, T., Wei, Y., Li, L., & Zhang, Y. (2019). Boundary perception guidance: A scribble-supervised semantic segmentation approach. In s. Kraus (Ed.), *Proceedings of the 28th International Joint Conference on Artificial Intelligence* (pp.3663–3669). IJCAI. doi:10.24963/ijcai.2019/508

Wang, L., Wu, J., Liu, X., Ma, X., & Cheng, J. (2022). Semantic segmentation of large-scale point clouds based on dilated nearest neighbors graph. *Complex & Intelligent Systems*, *8*(5), 3833–3845. doi:10.1007/s40747-021-00618-0

Wang, Y., Zhou, Q., Liu, J., Xiong, J., Gao, G., Wu, X., & Latecki, L. J. (2019). Lednet: A lightweight encoder-decoder network for real-time semantic segmentation. In *2019 IEEE International Conference on Image Processing (ICIP)* (pp. 1860–1864). IEEE. doi:10.1109/ICIP.2019.8803154

Xiao, X., Zhao, Y., Zhang, F., Luo, B., Yu, L., Chen, B., & Yang, C. (2023). BASeg: Boundary aware semantic segmentation for autonomous driving. *Neural Networks*, *157*, 460–470. doi:10.1016/j.neunet.2022.10.034 PMID:36434954

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Álvarez, J. M., & Luo, P. (2021). SegFormer: Simple and efficient design for semantic segmentation with transformers. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, & J. Wortman Vaughan (Eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021* (pp.12077–12090). NeurIPS. doi:10.48550/arXiv.2105.15203

Xu, G., Zhang, L., Chen, M., & He, B. (2021). Hybrid frontier detection strategy for autonomous exploration in multi-obstacles environment. *2021 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (pp. 1909–1915). IEEE. doi:10.1109/ROBIO54168.2021.9739463

Xu, J., Xiong, Z., & Bhattacharyya, S. P. (2023). PIDNet: A real-time semantic Segmentation network inspired by PID controllers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 19529–19539). IEEE. doi:10.1109/CVPR52729.2023.01871

Xu, Z., He, D., Vijayakumar, P., Gupta, B. B., & Shen, J. (2021). Certificateless public auditing scheme with data privacy and dynamics in group user model of cloud-assisted medical WSNs. *IEEE Journal of Biomedical and Health Informatics*, *27*(5), 2334–2344. doi:10.1109/JBHI.2021.3128775 PMID:34788225

Yang, D., Zhu, T., Wang, S., Wang, S., & Xiong, Z. (2022). LFRSNet: A robust light field semantic segmentation network combining contextual and geometric features. *Frontiers in Environmental Science*, *10*, 996513. doi:10.3389/fenvs.2022.996513

Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., & Sang, N. (2018). Learning a discriminative feature network for semantic segmentation. In *Proceedings of the 2018 IEEE conference on Computer Vision and Pattern Recognition* (pp. 1857–1866). IEEE. doi:10.1109/CVPR.2018.00199

Yuan, Y., Chen, X., & Wang, J. (2020). Object-contextual representations for semantic segmentation. In A. Vedaldi, H. Bischof, T. Brox, & J. M. Frahm (Eds.), Lecture notes in computer science: Vol. 12357. *Computer Vision–ECCV 2020* (pp. 173–190). Springer. doi:10.1007/978-3-030-58539-6_11

Yuan, Y., Xie, J., Chen, X., & Wang, J. (2020). Segfix: Model-agnostic boundary refinement for segmentation. In A. Vedaldi, H. Bischof, T. Brox, & J. M. Frahm (Eds.), Lecture notes in computer science: Vol. 12357. *Computer Vision–ECCV 2020* (pp. 489–506). Springer. doi:10.1007/978-3-030-58610-2_29

Zhang, F., Chen, Y., Li, Z., Hong, Z., Liu, J., Ma, F., Han, J., & Ding, E. (2019). Acfnet: Attentional class feature network for semantic segmentation. In *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision* (pp. 6797–6806). IEEE. doi:10.1109/ICCV.2019.00690

Zhang, W., Huang, Z., Luo, G., Chen, T., Wang, X., Liu, W., Yu, G., & Shen, C. (2022). TopFormer: Token pyramid transformer for mobile semantic segmentation. In *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 12073–12083). IEEE. doi:10.1109/CVPR52688.2022.01177

Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017). Pyramid scene parsing network. In *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition* (pp. 6230–6239). IEEE. doi:10.1109/CVPR.2017.660

Zhao, Y., Xiong, Z., Zhou, S., Peng, Z., Campoy, P., & Zhang, L. (2022). KSF-SLAM: A key segmentation frame based semantic SLAM in dynamic environments. *Journal of Intelligent & Robotic Systems*, *105*(3), 3. Advance online publication. doi:10.1007/s10846-022-01613-4

Zhen, M., Wang, J., Zhou, L., Li, S., Shen, T., Shang, J., Fang, T., & Long, Q. (2020). Joint semantic segmentation and boundary detection using iterative pyramid contexts. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 13663–13672). IEEE. doi:10.1109/CVPR42600.2020.01368

Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., & Torralba, A. (2017). Scene parsing through ADE20K dataset. In *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition* (pp. 5122–5130). IEEE. doi:10.1109/CVPR.2017.544

Zhou, Q., Qiang, Y., Mo, Y., Wu, X., & Latecki, L. J. (2022). Banet: Boundary-assistant encoder-decoder network for semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, *23*(12), 6980. doi:10.1109/TITS.2022.3194213

APPENDIX

Table 7. Parameters and symbols

Section	Symbols and Parameters	Meaning
Methods	F_s	Semantic Feature Map
	F_b	Boundary Information Feature Map
	F_{b1}	Boundary information feature map of the first stage
	F_{b2}	Boundary information feature map of the second stage
	F_{b3}	Boundary information feature map of the third stage
	F_{b4}	Boundary information feature map of the fourth stage
	F_{s1}	Semantic feature map of the first stage
	F_{s2}	Semantic feature map of the second stage
	F_{s3}	Semantic feature map of the third stage
	F_{s4}	Semantic feature map of the fourth stage
Ablation Study	FLOPs	Floating Point Operations Per Second
	mIoU	Mean Intersection over Union
	FPS	Frames Per Second