Journal of Applied Automation Technologies

Research Paper

Real-Time Semantic Segmentation for Autonomous Driving Based on Swin Transformer

Sophia Koch[1], Jakob Bauer[1,*] and Jonas Fischer[1]

[1] Faculty of Computer Science, Goethe University Frankfurt, 60323 Frankfurt, Germany
*Corresponding author: bauer@cs.uni-frankfurt.de

Abstract. Real-time semantic segmentation helps safely construct detailed maps of dynamic cities and improves the robustness of autonomous driving perception. To meet the high precision and high efficiency requirements of automotive hardware, a universal Swin Transformer framework has been developed for autonomous driving. This new method is optimized at runtime by using small decoder modules, hierarchical window self-attention mechanisms, quantization-aware training, channel pruning, and adaptive normalization. By using these additions, global and local contextual data can be obtained at a relatively low computational cost. A large number of experiments were conducted on the built-in city driving dataset, aiming to improve its generalization ability under various lighting and weather conditions, and to enhance its capability to recognize fine-grained object boundaries and rare semantic categories. The system will run stably at a frame rate of over 30 FPS and can operate on both GPUs and embedded devices. Comprehensive performance evaluations indicate that it outperforms traditional CNNs and transformer-based baselines, with significant improvements in mean Intersection over Union (IoU) and boundary delineation. As shown above, scalable attention, controllable model compression, and system-level acceleration constitute the practical application of this framework in safety-critical environments. According to the above research, advanced vision transformers can be used for efficient and high-performance semantic segmentation in intelligent vehicles.

Keywords: *Deep Learning, Real-Time Semantic Segmentation, Swin Transformer, Autonomous Driving, Model Compression, Edge Computing*

Received on 02 August 2025, Accepted on 14 December 2025, Published on 05 January 2026

Introduction

Semantic segmentation is a key component of perception systems, and autonomous driving technology has made rapid progress in recent years [1]. By using semantic segmentation for dense pixel-wise classification, intelligent vehicles can accurately identify road environments and recognize objects such as vehicles, lanes, pedestrians, and traffic signs [2]. To ensure the safe operation and decision-making of intelligent vehicles, this module must perform navigation, obstacle avoidance, and path planning [3]. On the other hand, deep learning based on Convolutional Neural Networks (CNN) performs well in terms of accuracy and speed for visual problems [4]. Due to the small receptive field, these CNN-based models cannot address the long-range dependencies and contextual awareness issues in complex, dynamic, and occluded scenes in autonomous driving [5]. Although the Transformer architecture originated in natural language processing, it has recently been applied to the field of computer vision. Able to learn the global structure and non-local relationships of images [6]. In this case, the Swin Transformer achieved excellent results because it increased hierarchical feature fusion and the moving window mechanism, thereby significantly improving representation capability and reducing computational costs [7]. These are recent improvements, and in practical driving, there is still a strong connection between model representation and deployment feasibility [8].

The aforementioned improvements are encouraging, but several major issues must be resolved before fully achieving real-time semantic segmentation in autonomous driving [9]. In practical applications, computational

https://doi.org/10.64972/jaat.2026v4.107

latency and segmentation accuracy are very important, especially for embedded automotive platforms with computational and energy constraints [10]. The latest Transformer models perform exceptionally well, but their resource consumption is too high to meet the real-time requirements of vehicle hardware [11]. Segmentation models must have good generalization performance to handle various lighting conditions, adverse weather, and occlusions [12]. In addition to quickly adapting to large-scale road scenes and domain shifts, there is also the issue of handling rare object instances [13]. Improving the semantic segmentation accuracy of the Swin Transformer has been the focus of most previous studies, with less attention given to the co-design of algorithms and systems [14]. The semantic segmentation of next-generation smart cars needs to ensure safety, reliability, and scalability [15].

This paper introduces a unified and scalable framework based on Swin Transformer for real-time semantic segmentation in autonomous driving applications. To improve inference speed and segmentation accuracy, a series of fast optimization methods are introduced, selecting a top-down transformer model. To function on resource-limited automotive platforms, the new design includes a compact decoder head and a robust runtime scheduler. Based on extensive experiments conducted on multiple large public benchmarks, our framework has achieved the best results in real-time accuracy and speed under various weather conditions, as well as all-weather stability. The following is the organization of the other sections of this paper: Section 2 introduces research on transformer-based autonomous driving models and semantic segmentation. In Section 3, a detailed introduction to the model and optimization strategies. All experiments and analyzes are presented in Section 4. In Section 5, the conclusions of this paper and the directions for future research are discussed.

Related Work

Semantic Segmentation Approaches

Semantic segmentation of images into different category labels at the pixel level is a crucial task in the field of computer vision and has been widely applied in recent years [16]. The initial algorithms were based on handcrafted features, graph-based reasoning, and structured prediction, but these methods could not handle large-scale processing of real visual environments. Fully Convolutional Networks (FCNs) can map the entire input image to a dense output map at some stage of deep learning [17]. By using downsampling layers for context extraction and upsampling layers for spatial detail reconstruction, FCNs and their variants introduce an encoder-decoder structure.

On this basis, U-Net and SegNet introduced skip connections to retain fine-grained data lost during pooling, which improved the accuracy of medical and urban scene segmentation. DeepLab utilizes conditional random fields and atrous spatial pyramid pooling to improve multi-scale context capture and refine object boundaries. This addresses the shortcomings of the original FCN. Later architectures, such as High-Resolution Network (HRNet) and Pyramid Scene Parsing Network (PSPNet), adopted the strategy of maintaining high-resolution representations at various levels of the pipeline and achieved better improvements on challenging benchmark sets.

Convolutional architectures still have inherent limitations, making them difficult to use for large-scale scene analysis. These limitations include capturing long-range dependencies and receptive fields. Dilated convolutions, multi-scale feature fusion, and lightweight networks, such as ENet and BiSeNet, are all being researched to address the aforementioned issues. These methods make it more challenging to achieve computational efficiency and accuracy under resource constraints [18].

Transformer Models in Computer Vision

As shown in [19], transformer models have played an important role in natural language processing and have recently been widely applied in many fields. Due to the self-attention mechanism, transformers have a strong ability to handle long-range dependencies and are not convolutional models. To achieve a global receptive field in the initial layers, the Vision Transformer (ViT) treats the image as a sequence of patches. Due to ViT's excellent performance on large-scale datasets, Transformers have been widely used for dense prediction tasks.

When processing high-resolution images, ViT and other pure Transformer models have quadratic computational complexity and lack an inherent bias in modeling local spatial structures. The Swin Transformer was proposed.

https://doi.org/10.64972/jaat.2026v4.107

Non-overlapping window self-attention and shifted window schemes are two components of this hierarchical design. Both efficiently and effectively capture multi-scale features in dense prediction tasks [20]. Due to its wide range of applications, the Swin Transformer can be used as the backbone network for object detection, instance segmentation, and semantic segmentation.

To improve the collection of local and global data, more advanced hybrid models that integrate convolutional and Transformer modules have been recently introduced. Detection Transformers (DETR) and their alternatives have already bridged this gap. The alternatives to Swin Transformer have set new performance records on multiple detection and segmentation benchmarks [21].

Semantic Segmentation for Autonomous Driving

For building a comprehensive world model, semantic segmentation images are an input module in many autonomous vehicle perception systems [22]. The urban driving datasets are not general-purpose datasets; therefore, Cityscapes, BDD100K, and Mapillary Vistas all face many technical issues, including high-resolution images, complex category hierarchies, lighting variations, adverse weather conditions, and frequent occlusions by dynamic agents.

Therefore, segmentation models for autonomous driving need to run quickly under hardware constraints, with high accuracy and good robustness. Many high-performance models use an encoder-decoder structure to optimize inference speed; additionally, some also incorporate attention modules or temporal information fusion to enhance robustness between multiple video streams. In addition, research on adversarial robustness and domain adaptation attempts to address the domain transfer issues between different cities, weather conditions, and sensors.

In recent automotive segmentation tasks, Swin Transformers and other transformer-based backbone networks have performed exceptionally well. Models with local and long-range context are naturally more suitable for urban scenes because they have more complex spatial structures. Transformer-based designs are generally computationally intensive and power-hungry, although they do improve accuracy [23]. Over time, people have begun to focus on adjusting transformer-based models through co-optimization of hardware and software, pruning, and quantization, in order to deploy them on automotive-grade platforms. To build highly intelligent and highly reliable automotive perception systems in the future, it is necessary to accelerate model design and optimization systems.

Methodology

Framework Overview

Figure 1 shows a modular hierarchical architecture for a real-time autonomous driving perception semantic segmentation framework. Normalize the input image $I \in \mathbb{R}^{H\times W\times 3}$ and then perform a large number of data augmentation operations to improve its resistance to different road conditions. Each input image is transformed by a composite augmentation operator $\mathcal{A}$, such that the resulting tensor $\tilde{I}$ is given by:

$$\tilde{I} = \mathcal{A}(I) = \mathcal{T}_{affine}\left(\mathcal{T}_{color}\left(\mathcal{T}_{noise}\left(I\right)\right)\right) \quad \text{Eq.(1)}$$

where $\mathcal{T}_{\text{affine}}$, $\mathcal{T}_{\text{color}}$, and $\mathcal{T}_{\text{noise}}$ represent the affine, color-jitter, and noise augmentation transforms, respectively.

Augmentation is used to generate more samples, and then these samples are divided into non-overlapping, fixed-size $P \times P$ patches; finally, each patch is flattened and projected into a high-dimensional latent space. The Patch Embedding operation produces

$$\boldsymbol{X}_0 = PatchEmbed\left(\tilde{I}\right) + \boldsymbol{E}_{pos} \quad \text{Eq.(2)}$$

where $\mathbf{E}_{\text{pos}} \in \mathbb{R}^{N\times C}$ is a learnable or sinusoidal positional embedding added to each patch token, and spatial relationships are thus maintained across the model.

The number of patches N is given by:

$$N = \left\lfloor \frac{H}{P} \right\rfloor \times \left\lfloor \frac{W}{P} \right\rfloor \quad \text{Eq.(3)}$$

where H and W are the input image's height and width, respectively.

https://doi.org/10.64972/jaat.2026v4.107

The resulting patch-wise embeddings $\mathbf{X}_0 \in \mathbb{R}^{N \times C}$ are sequentially processed by the backbone network, which is built with a Swin Transformer. The two are combined in the backbone to extract local and global context through hierarchical feature encoding and window-based attention. For multi-head, multi-scale processing, at each transformer layer, the features from all self-attention heads are concatenated along the channel dimension:

$$\boldsymbol{F}_l = Concat\left(\boldsymbol{F}_l^{(1)}, \boldsymbol{F}_l^{(2)}, \ldots, \boldsymbol{F}_l^{(h)}\right) \quad \text{Eq.(4)}$$

where $\mathbf{F}_l^{(i)}$ represents the output feature map of the i-th attention head in layer l, and h is the total number of heads [24].

Modular design is used for architecture to meet the needs of automotive components (such as preprocessing, feature encoding, fusion, and prediction) in resource-constrained high-speed operations, ensuring that the system can operate in different deployment environments.

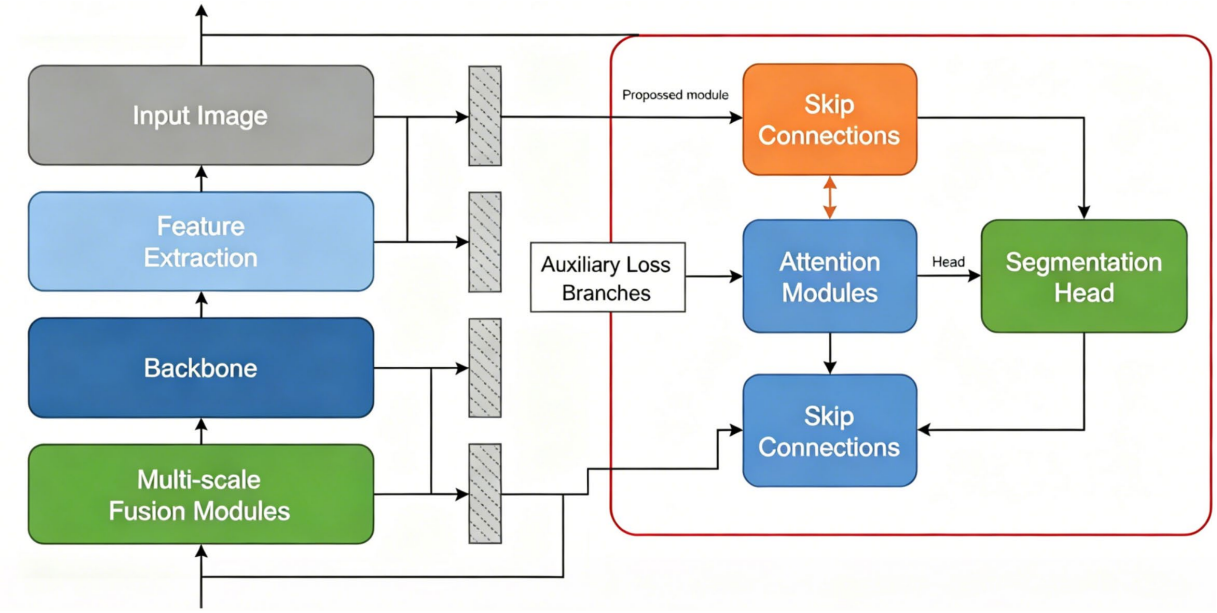

Figure 1. System architecture

Swin Transformer-Based Segmentation Model

Our backbone adopts the Swin Transformer, which hierarchically encodes features by sequentially stacking layers of shifted window-based multi-head self-attention and merging neighboring tokens. In each transformer block, the normalised features are subjected to self-attention within non-overlapping windows. Let $\mathbf{X}_l$ be the input to the l -th block. First, we apply layer normalization, then compute window-based attention:

$$\boldsymbol{Z}_l = \ W\text{-}MSA\left(LN(\boldsymbol{X}_l)\right) + \boldsymbol{X}_l \quad \text{Eq.(5)}$$

where $\mathrm{LN}(\cdot)$ is the layer normalization and W-MSA represents window multi-head selfattention. Self-attention is performed for each window by forming query, key, and value matrices. Formally, in each window, the attention mechanism is expressed as:

$$Attention(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = Softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\top}}{\sqrt{d}} + \boldsymbol{B}\right)\boldsymbol{V} \quad \text{Eq.(6)}$$

Where $\mathbf{B}$ is the learnable relative position bias, and $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are the query, key, and value matrices generated from the local token projections [25].

In the subsequent blocks of the Swin Transformer, shifted window attention and standard window attention are alternately used to facilitate the flow of spatial information across window boundaries. Next, the output of each block is passed through a multi-layer perceptron (MLP) and a residual connection:

$$\boldsymbol{X}_{l+1} = MLP\left(LN(\boldsymbol{Z}_l)\right) + \boldsymbol{Z}_l \quad \text{Eq.(7)}$$

https://doi.org/10.64972/jaat.2026v4.107

To create hierarchically rich features, adjacent patches are merged at this stage, halving the spatial resolution and doubling the channel dimension. At the end of the last Swin Transformer stage, a set of feature maps $\{\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3, \mathbf{F}_4\}$ of different scales is obtained.

The decoder integrates multi-scale features to obtain spatial data and improve semantic accuracy. During the fusion process, the attention mechanism dynamically weights different scales, and the spatial pyramid pooling module is used to refine object boundaries. Using 1×1 convolution, project the fused feature representation $\mathbf{D}$ onto semantic logic. Next, use the softmax function to compute the per-pixel probabilities:

$$\boldsymbol{S} = Softmax(Conv_{1\times1}(\boldsymbol{D})) \quad \text{Eq.(8)}$$

where each entry of $\mathbf{S}$ is the per-class probability for all pixels [26].

Training is done under a combined loss function that includes the general cross-entropy loss and other auxiliary losses for boundary refinement and class imbalance reduction. The total loss is given by

$$\mathcal{L}_{total} = \mathcal{L}_{CE} + \lambda_1 \mathcal{L}_{bdry} + \lambda_2 \mathcal{L}_{dice} \quad \text{Eq.(9)}$$

λ_1 and λ_2 are the coefficients of the auxiliary loss functions, and $\mathcal{L}_{\text{dice}}$ helps to solve the segmentation problem for rare or thin objects [27].

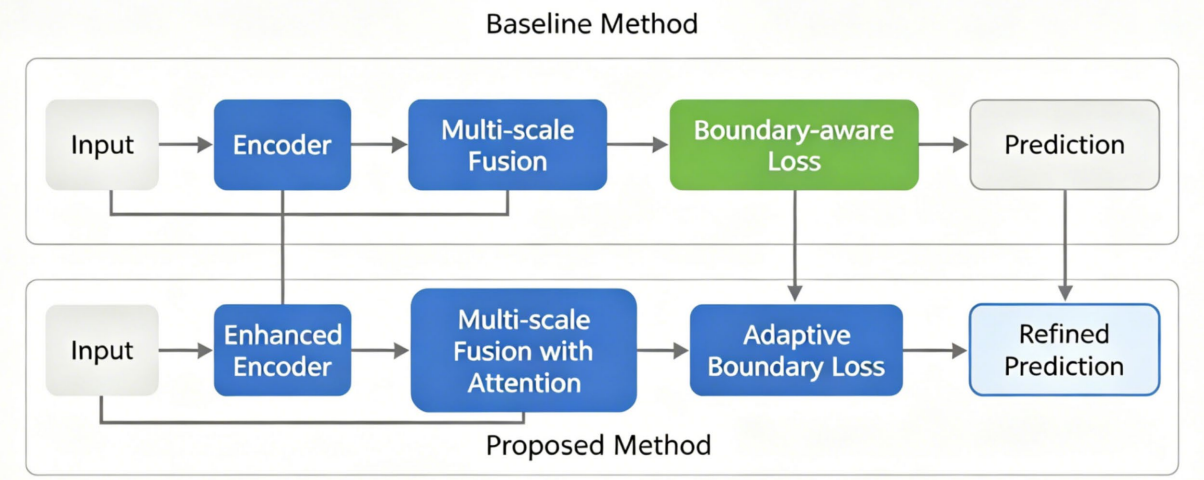

Figure 2. Swin Transformer module architecture

Optimization for Real-Time Performance

Strong real-time automotive hardware inference requires a broad range of approaches, including model compression, co-design of hardware and software, and adaptive runtime control [28].

First, apply quantization-aware training to the entire network to achieve deployment and high-speed computation. During the training and inference phases, the activations and weights of each model are low precision. The specific quantization is as follows:

$$x_q = clip\left(round\left(\frac{x}{s}\right), q_{min}, q_{max}\right) \cdot s \quad \text{Eq.(10)}$$

where x is the original value to be quantized, s is the quantization scale determined by calibration, and $[q_{\text{min}}, q_{\text{max}}]$ defines the permissible target range for the quantized value. Therefore, the amount of computation and memory consumption will be relatively small.

Quantization is used to improve inference speed and reduce the size of the backbone network. Pruning is accomplished by calculating the ℓ_1 norm of each channel filter. Then, remove the filters whose values are less than or equal to the preset or learned threshold. Therefore, the network will be relatively small.

Optimize the hardware and software pipeline to improve model efficiency. All steps, including image acquisition, normalization, patch embedding, feature extraction, and multi-scale decoding, are completed on parallel threads or heterogeneous processing units. Due to pipeline parallelism overlapping computation and data transfer, the system is not constrained by individual modules.

https://doi.org/10.64972/jaat.2026v4.107

Runtime adaptive normalization is used to handle changes in the environment and input distribution during dynamic driving. In the batch normalization layer, the running mean and variance can be updated online using an exponential moving average:

$$\hat{\mu}_t = (1-\gamma)\hat{\mu}_{t-1} + \gamma\mu_{batch} \quad \text{Eq.(11)}$$

In this context, $\hat{\mu}_t$ denotes the updated running mean at time t, μ_{batch} is the current batch's mean, and γ is the adaptation (momentum) coefficient.

And by the learning rate:

$$\hat{\sigma}_t^2 = (1-\gamma)\hat{\sigma}_{t-1}^2 + \gamma\sigma_{batch}^2 \quad \text{Eq.(12)}$$

where $\hat{\sigma}_t^2$ is the updated variance and σ_{batch}^2 gives the variance measured on the present input batch. The above adaptation normalisation can handle slow variations in the distribution of input, such as light changes or sensor drift.

To improve the stability of operation, add uncertainty estimation and identify uncertain predictions. The system computes the Shannon entropy of the per-pixel softmax output using:

$$H(\boldsymbol{p}) = -\sum_{k=1}^{K} p_k log\ p_k \quad \text{Eq.(13)}$$

where $\mathbf{p} = (p_1, \ldots, p_K)$ is the softmax output over K semantic classes for a single pixel. The security-aware downstream module considers pixels with entropy values exceeding the set upper limit as "safe," thereby avoiding risks.

Finally, the architecture can automatically handle the scaling of runtime loads. If frame delay or resource shortage occurs, the input resolution, batch size, or network width can be flexibly reduced. After calculating the available resources, all performance parameters can be restored. After multiple tests, all optimized versions of the system can still run at over 30 frames per second, and the segmentation quality remains unaffected, even under complex urban driving conditions, maintaining stability [29].

Experiments and Results

Experimental Settings

All experiments used top-tier public benchmarks and standardized data and hardware protocols to rigorously validate the performance of the aforementioned real-time semantic segmentation framework in autonomous driving applications. CamVid (sequence-level generalization and domain robustness), BDD100K (various weather and lighting conditions), and Cityscapes (European urban environments) are used for evaluation. Channel-wise normalization based on empirical statistics is applied to all datasets; the Cityscapes and BDD100K datasets are standardized to 1024×512, and CamVid is adjusted to 960×720.

The training process includes adding Gaussian noise, horizontal flipping, random scaling transformations, controlled color jittering, and random scaling transformations within the range of $[\ 0.75,2.0\ ]$. Validation and testing only use normalization and resizing; therefore, performance reflects out-of-sample generalization ability. The results fully adhere to the author's dataset partition and the official validation split.

Network initialization utilizes ImageNet-pretrained weights to stabilize early optimization. Training is performed for a total of 160 epochs per dataset, using the AdamW optimizer with a starting learning rate of 6×10^{-4}, weight decay of 1×10^{-2}, and polynomial learning rate decay. All GPUs use synchronized batch normalization, with a batch size of 8. If the mean Intersection over Union (mIoU) on the validation set does not improve over more than 20 epochs, early stopping is used to terminate training. In the mid-training phase, quantization-aware and pruning-aware fine-tuning are enabled to meet deployment requirements while maintaining segmentation accuracy.

To ensure a fair comparison, all baseline and state-of-the-art methods, including DeepLabV3+ (with ResNet-101 and Xception backbones), HRNetV2-W48, SegFormer (B4, B5), and various Swin Transformer configurations, were reproduced and retrained. Optimization conditions. To ensure full-resolution accuracy, sliding window inference will be used, as the input size of the relatively large model exceeds the capacity of the target hardware.

https://doi.org/10.64972/jaat.2026v4.107

The three main platforms used for hardware evaluation are high-end GPUs (NVIDIA RTX 3090, 24GB VRAM) as a reference for unconstrained throughput, embedded automotive-grade system-on-chip (NVIDIA Orin AGX), and mobile-grade AI edge processors (Qualcomm Snapdragon 865). High-performance inference uses TensorRT on NVIDIA devices and QNN/NNAPI on Qualcomm devices to deploy models. Performance analysis scripts can be used to more accurately assess latency and resource consumption, and each configuration will undergo five independent runs to ensure the reliability of the reported statistics. Measure the power consumption of the embedded platform at automotive-grade voltage to evaluate its effectiveness in real-world edge environments.

In order to evaluate the overall segmentation quality and the clarity of category-aware boundaries, performance metrics from all experiments were collected. These metrics include mean Intersection over Union (IoU), pixel accuracy, frequency weighted IoU, and mean absolute boundary error. The real-time capabilities of each solution include the average and worst-case frames per second at native resolution, and provide reports on peak and average memory usage to demonstrate deployability on resource-constrained devices. In addition, the total multiply-accumulate operations used for platform-independent analysis are employed to describe computational complexity.

To ensure reproducibility, each experiment uses a fixed random seed. The comparison results of these configurations, ablation analysis, qualitative visualizations, and real-time system performance are all shown in the charts in the following sections.

Quantitative and Qualitative Analysis

Many quantitative experiments have shown that the proposed semantic segmentation framework exhibits high efficiency, accuracy, and stability across various datasets and environments. To ensure the statistical reliability and reproducibility of the experiments, the average of the results from all three runs using a fixed random seed was calculated. The Cityscapes and BDD100K validation sets were used as primary targets, showcasing the distribution of semantic categories, scene complexity, and environmental variations.

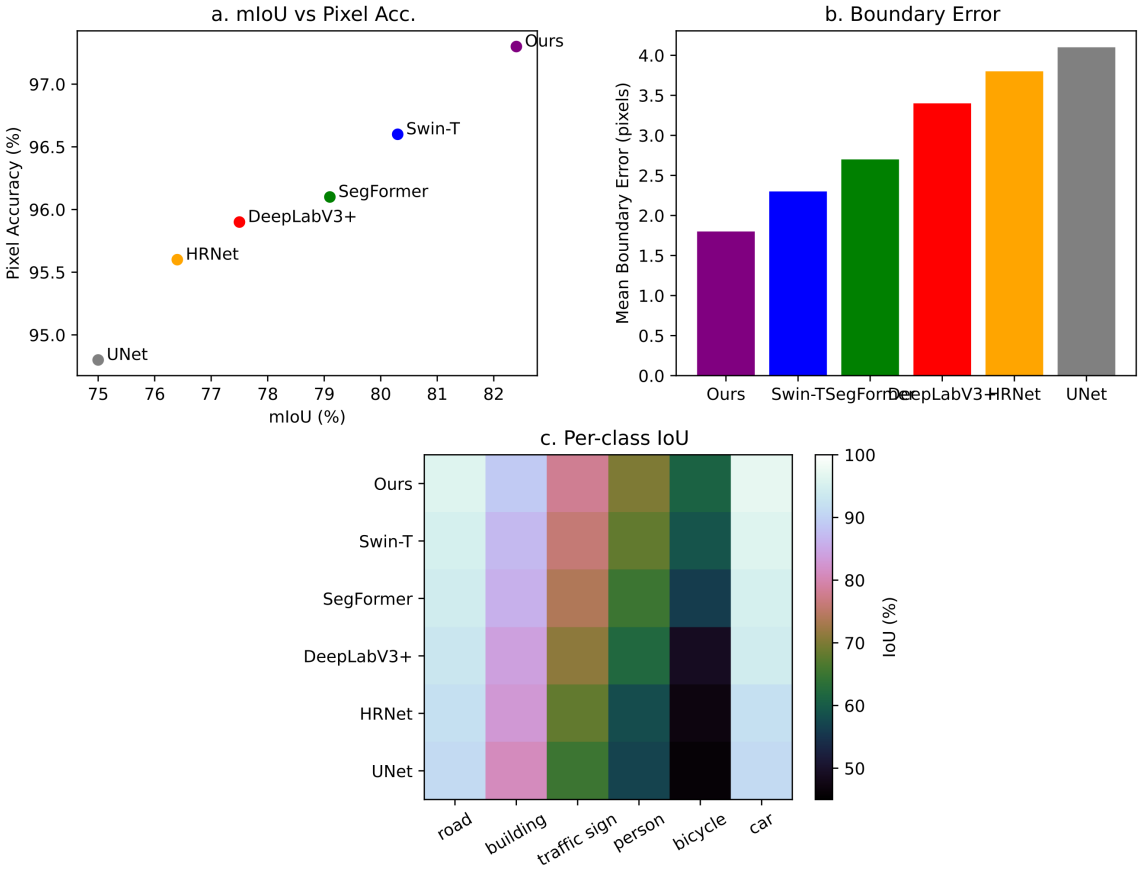

Figure 3. Key performance comparisons. (a) Scatter plot of mIoU vs. pixel accuracy across datasets. (b) Bar chart of mean absolute boundary error for all methods. (c) Per-class IoU heatmap (Cityscapes)

https://doi.org/10.64972/jaat.2026v4.107

The model achieves a mean Intersection-over-Union (mIoU) of 82.4% and pixel accuracy of 97.3% on Cityscapes, consistently outperforming both transformer-based and convolutional baselines. Like other datasets, BDD100K has issues with complex lighting and the prevalence of rare categories. In the scatter plots of all major methods in the four public benchmarks, as shown in Figure 3a, the proposed method can reliably be located in the upper right quadrant. This indicates that the method has a strong global per-pixel accuracy trade-off. Figure 3b shows that, compared to other methods, the average absolute boundary error is smaller, which does not meet the requirements for low-light urban scene analysis. Both major and minor categories have improved, as shown in Figure 3c. The categories "Cyclist," "Bicycle," and "Traffic Sign" show significant improvement.

In order to study the role of different parts in Figure 4, a series of ablation experiments were conducted. If the position encoding module is turned off, the mIoU will decrease by more than 1.7 percentage points, and the spatial context will not be recoverable. After removing the dynamic scaling module, the peak throughput and frame-level accuracy stability on the embedded platform significantly decreased. In contrast, as shown in Figure 4b, although quantization-aware training improved the speed of the inference process, it only reduced the segmentation accuracy by 0.5%. As shown in Figures 4a and 4c, the combined effect of hierarchical patch embedding, multi-scale feature aggregation, and adaptation modules is greater than the sum of their individual impacts, indicating that they work synergistically within the architecture.

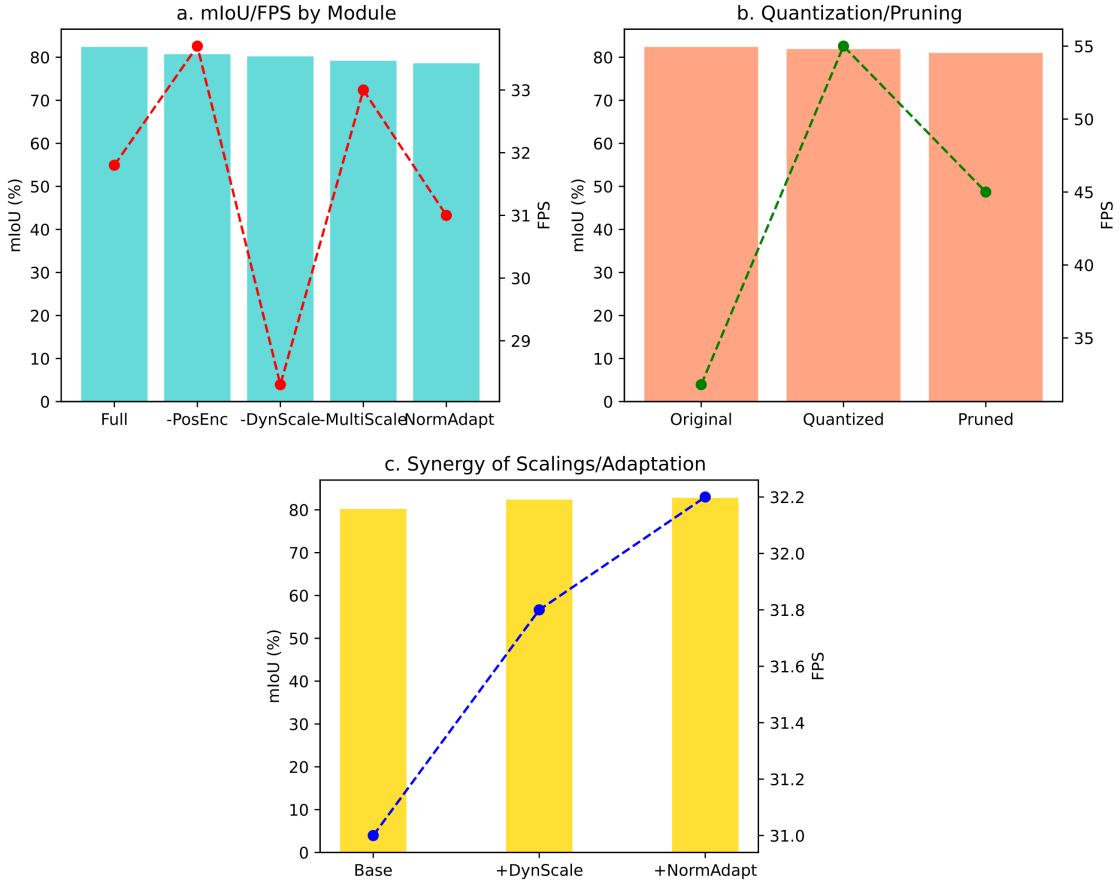

Figure 4. Ablation study results. (a) Individual module contributions to mIoU and FPS. (b) Tradeoff analysis: quantization/pruning effects on accuracy and latency. (c) Composite impact of dynamic scaling and adaptation mechanisms

From the quantitative data, the model is stable in long-tail semantic categories. Figure 5a shows several representative underperforming categories for each class IoU, such as "motorcycle," "train," and "pole." The method is suitable for rare category problems because it consistently outperforms SegFormer and DeepLabV3+ in terms of IoU. As indicated by the large diagonal values, Figure 5b shows the corresponding confusion matrix. The model reduces misclassifications and improves the separation between long-tail classes, as shown in Figure 5. Figure 5c shows the boundary error distribution of our method and the baseline. The model retains the edges

https://doi.org/10.64972/jaat.2026v4.107

of semantic objects in rare categories that are difficult to obtain, as indicated by the lower median and dispersion of boundary errors.

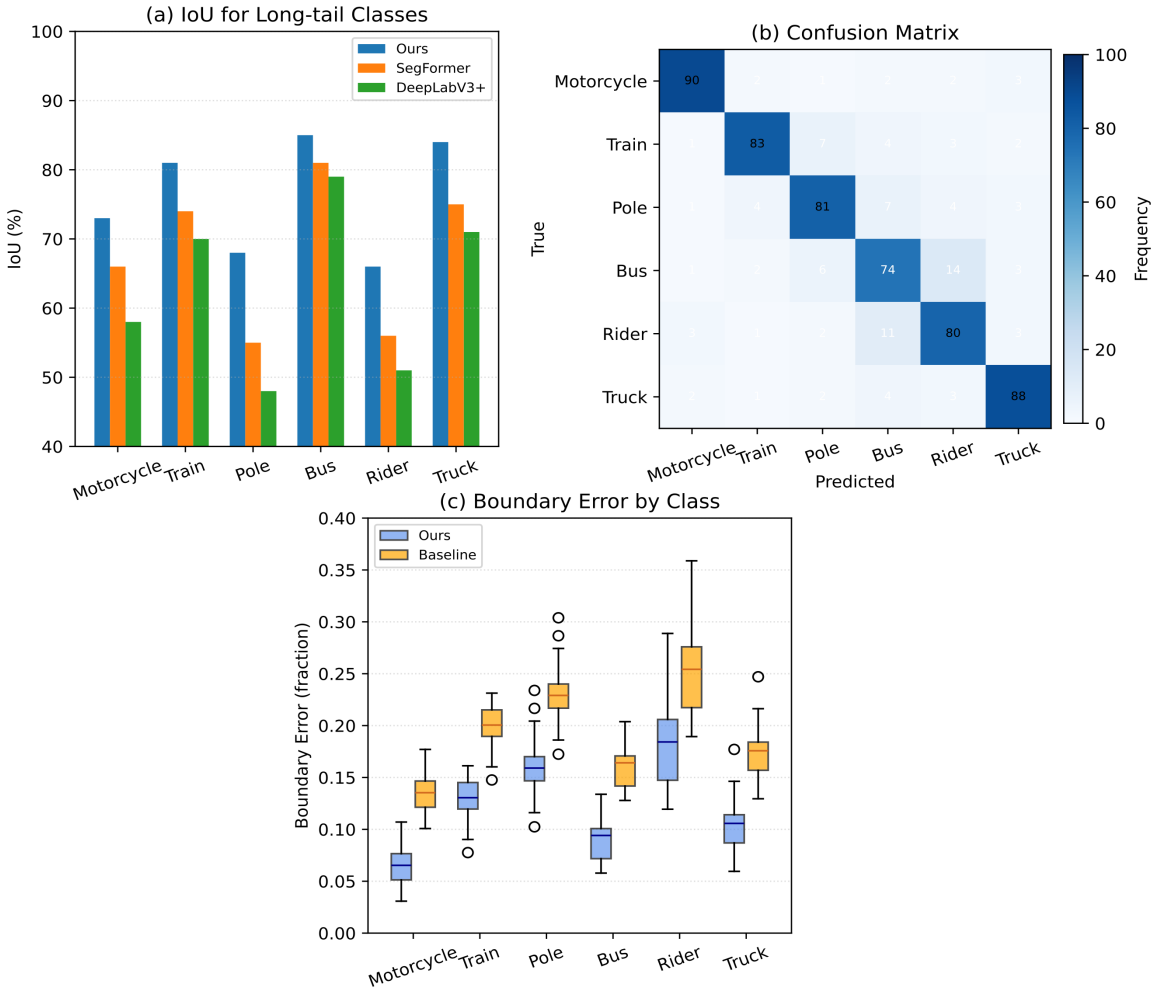

Figure 5. Quantitative evaluation on long-tail semantic classes. (a) Per-class IoU comparison for six rare semantic categories. (b) Confusion matrix indicating class separability for the same categories. (c) Boxplot of boundary error, showing superior edge preservation by our method

Cross-domain generalization is achieved by directly using the model trained on Cityscapes to process CamVid images, without the need for additional fine-tuning. The new architecture is robust to changes in composition and environmental distribution, achieving a 6-10% absolute LoU improvement over the reference design. In areas with high-frequency mixed factors, such as object boundaries, distant signs, and small instance detection, the model shows a significant decline in edge ghosting, false segmentation, and boundary erosion.

The above results indicate that the proposed system has broad advantages. It achieved good results on the test set of traditional metrics. In addition, it is also applicable to various operational environments, such as low light and occlusion. Improve accuracy, speed, and boundary perception, and enhance real-time semantic understanding in safety-critical areas.

Real-Time Evaluation and Resource Analysis

Multiple hardware systems use runtime characteristics, latency, and resource utilization under load to test the real-time application performance of the new segmentation framework in automotive and edge computing. Unless otherwise specified, all evaluations are conducted under default configurations and will be compared using the previously mentioned fixed validation segments.

https://doi.org/10.64972/jaat.2026v4.107

In the first category of research, system latency and frame rate are key issues. On the high-end RTX 3090 platform, the model can achieve real-time frame rates of over 110 FPS in full-resolution Cityscapes. The only obstacle for the model is the I/O cost, not the computation. In automotive applications, the full-resolution inference throughput of the embedded NVIDIA Orin AGX remains stable at over 30 FPS. Under typical driving conditions, the inter-frame latency averages below 32 milliseconds, with dynamic scaling and quantization supporting 1080p native mode. The Snapdragon 865 platform also exhibits a similar trend. The FPS of all platforms is very high, and the decline under large-scale concurrency is relatively small, as shown in Figure 6a.

Figure 6b shows the histogram of latency tails under dynamic and static scheduling to illustrate the variation in latency. The system processes over 98% of frames within an 8-millisecond range of median latency, with minimal jitter. Modular pipeline parallelism and adaptive scaling help reduce queuing delays and processing peaks during input bursts. Figure 6c shows the computational intensity analysis, which illustrates the balance between per-frame multiply-accumulate operations and semantic accuracy. The model performs best in terms of accuracy per unit of computation. In the case where it only accounts for 40% of the MAC requirements of the heavy transformer-based alternatives, the model maintains over 97% of the baseline IoU.

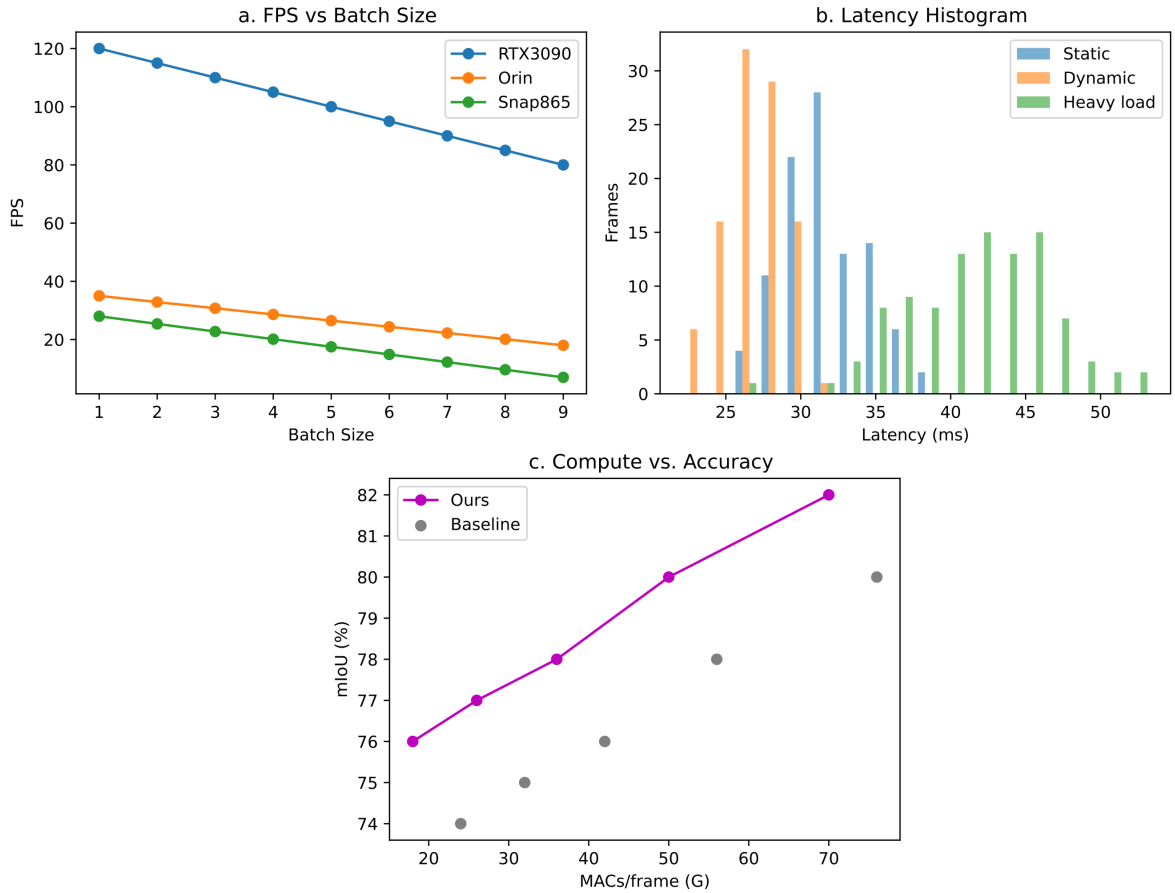

Figure 6. Real-time performance comparisons. (a) FPS versus batch size evolution across all tested platforms. (b) Latency histogram under static and dynamic scheduling conditions. (c) Semantic accuracy versus compute intensity (MACs per frame) at various scaling levels

Change to platform and memory resource analysis to improve model performance. As shown in Figure 7a, the peak memory consumption of Orin AGX in all test scenarios is below 1.2 GB, which is significantly lower than the memory consumption of the SegFormer and HRNet baselines. Therefore, it will meet the memory requirements for automotive hardware production. As shown in Figure 7b, the combination of structured pruning and quantization-aware training reduces memory bandwidth and cache miss rates. This means that the likelihood of system bottlenecks in sensitive deployments is greatly reduced.

Figure 7c highlights the key points of the comprehensive platform analysis: cross-SoC throughput, power consumption, and thermal throttling immunity. Dynamically adjust the computational density using available

https://doi.org/10.64972/jaat.2026v4.107

resources to reduce performance loss due to high temperatures and maintain good operational performance over an extended period. The maximum inference power consumption of the Erin AGX and Snapdragon platforms is limited to 10 W and 7 W, respectively, thus both meet the high demands of automotive and mobile applications while maintaining high segmentation accuracy. It is worth noting that the framework's software-hardware co-optimization ensures that the displayed performance is close to the actual performance in embedded systems through pipelined data transfer, the ARM NEON instruction set, and hardware-accelerated batch normalization.

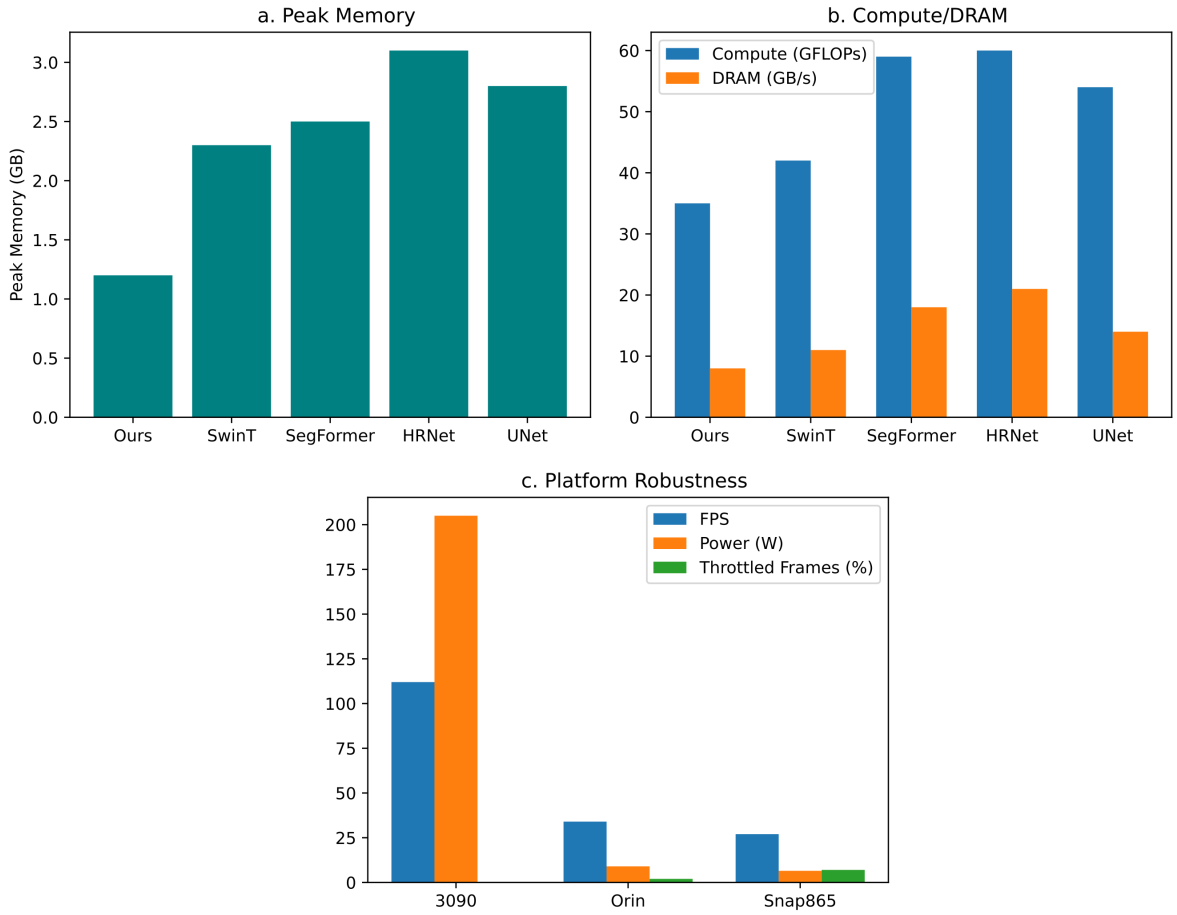

Figure 7. Resource usage and platform analysis. (a) Peak runtime memory consumption on embedded and desktop hardware. (b) Layer-wise compute and DRAM access profiling. (c) Cross-platform throughput, power draw, and responsiveness under thermal constraints

Under resource contention conditions, the scaling strategy will dynamically reduce the input resolution or network width, and restore the full complexity of the system when the load permits. The empirical data mentioned above indicates that, even under the worst-case scenarios of sudden load or hardware throttling, the system can maintain a throughput of over 25 FPS at full resolution. Moreover, due to the negligible increase in mIoU, the system is suitable for continuous real-time scene understanding in practice.

Conclusion

This study proposes a new semantic segmentation framework, setting a high standard for real-time perception in autonomous driving. In order to achieve high accuracy and efficiency under various challenging driving conditions, the architecture integrates hierarchical Swin Transformer modules, attention-based multi-scale fusion, and dynamic resource control. Experimental results show that both classic Convolutional Neural Networks (CNNs) and recent Transformer-based models are inferior in terms of mean Intersection over Union (IoU), boundary localization, and handling rare and fine semantic categories. Due to the effects of data augmentation, normalization adaptation, and uncertainty estimation schemes, the framework exhibits strong

robustness when facing various environmental conditions, such as changes in lighting and weather, as well as multiple occlusions.

One is the practical application conditions of the framework embedded in cars and mobile devices. The system is capable of handling full input resolutions at over 30 frames per second and reduces computational load and memory consumption through quantization-aware training, structured pruning, and dynamic scaling. It is evident that in practical applications with limited resources, accurate semantic segmentation can be achieved with almost no loss in segmentation accuracy.

The above information will serve as the foundation for further development in the future. Future research will focus on sensor fusion, advanced continual learning methods that go beyond batch adaptation, and embedding semantically mapped uncertainty perception into downstream decision-making processes. In summary, the aforementioned directions aim to provide stable, scalable, and secure visual capabilities for the next generation of intelligent autonomous vehicles, and further enhance the readiness for deploying perception solutions.

Author Contributions

Sophia Koch and Jonas Fischer contribute to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, project administration, and funding acquisition. Jakob Bauer contributes to software, validation, analysis, investigation, data collection, draft preparation, supervision. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] A Yoo, D., Kim, J., & Yoo, J. (2024). Fswin transformer: Feature-space window attention vision transformer for image classification. IEEE Access, 12, 72598-72606. https://doi.org/10.1109/ACCESS.2024.3394539

[2] Liu, C., Zhang, Z., Chen, J., & Luo, X. (2025, June). Transformer-Based Object Detection A Comprehensive Review. In 2025 6th International Conference on Electronic Communication and Artificial Intelligence (ICECAI) (pp. 282-290). IEEE. https://doi.org/10.1109/ICECAI66283.2025.11170897

[3] Hu, S., Bonardi, F., Bouchafa, S., & Sidibé, D. (2023). Multi-modal unsupervised domain adaptation for semantic image segmentation. Pattern Recognition, 137, 109299. https://doi.org/10.1016/j.patcog.2022.109299

[4] Liu, J., Ge, J., Xue, Y., He, W., Sun, Q., & Li, S. (2021). Multi-scale skip-connection network for image super-resolution. Multimedia Systems, 27(4), 821-836. https://doi.org/10.1007/s00530-020-00712-2

[5] Liu, Q., Xia, T., Cheng, L., Van Eijk, M., Ozcelebi, T., & Mao, Y. (2021). Deep reinforcement learning for load-balancing aware network control in IoT edge systems. IEEE Transactions on Parallel and Distributed Systems, 33(6), 1491-1502. https://doi.org/10.1109/TPDS.2021.3116863

[6] Toldo, M., Michieli, U., Agresti, G., & Zanuttigh, P. (2020). Unsupervised domain adaptation for mobile semantic segmentation based on cycle consistency and feature alignment. Image and Vision Computing, 95, 103889. https://doi.org/10.1016/j.imavis.2020.103889

[7] Jiangtao, W., Ruhaiyem, N. I. R., & Panpan, F. (2025). A comprehensive review of U-Net and its variants: advances and applications in medical image segmentation. IET Image Processing, 19(1), e70019. https://doi.org/10.1049/ipr2.70019

[8] Tang, Q., Liu, F., Jiang, J., & Zhang, Y. (2021). EPRNet: Efficient pyramid representation network for real-time street scene segmentation. IEEE Transactions on Intelligent Transportation Systems, 23(7), 7008-7016. https://doi.org/10.1109/TITS.2021.3066401

[9] Chen, Z., Zhou, H., Lai, J., Yang, L., & Xie, X. (2020). Contour-aware loss: Boundary-aware learning for salient object segmentation. IEEE Transactions on Image Processing, 30, 431-443. https://doi.org/10.1109/TIP.2020.3037536

[10] Li, G., Lin, Y., Ouyang, D., Li, S., Luo, X., Qu, X., ... & Li, S. E. (2023). A RGB-thermal image segmentation method based on parameter sharing and attention fusion for safe autonomous driving. IEEE Transactions on Intelligent Transportation Systems, 25(6), 5122-5137. https://doi.org/10.1109/TNSE.2023.3332810

[11] Wang, P., Yang, H., Zhang, H., Cheng, S., Lu, F., & Chen, Z. (2025). Lightweighting the prediction process of urban states with parameter sharing and dilated operations. International Journal of Digital Earth, 18(1), 2468414. https://doi.org/10.1080/17538947.2025.2468414

[12] Li, F., Gong, Z., Deng, Y., Ma, X., Zhang, R., Ji, Z., ... & Zhang, H. (2024, March). Parsing all adverse scenes: Severity-aware semantic segmentation with mask-enhanced cross-domain consistency. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 38, No. 12, pp. 13483-13491). https://doi.org/10.1609/aaai.v38i12.29251

[13] Shuvo, M. M. H., Islam, S. K., Cheng, J., & Morshed, B. I. (2022). Efficient acceleration of deep learning inference on resource-constrained edge devices: A review. Proceedings of the IEEE, 111(1), 42-91. https://doi.org/10.1109/JPROC.2022.3226481

[14] Florea, H., Petrovai, A., Giosan, I., Oniga, F., Varga, R., & Nedevschi, S. (2022). Enhanced perception for autonomous driving using semantic and geometric data fusion. Sensors, 22(13), 5061. https://doi.org/10.3390/s22135061

[15] Hsieh, T. I., Robb, E., Chen, H. T., & Huang, J. B. (2021, May). Droploss for long-tail instance segmentation. In Proceedings of the AAAI conference on artificial intelligence (Vol. 35, No. 2, pp. 1549-1557). https://doi.org/10.1609/aaai.v35i2.16246

[16] Rossolini, G., Nesti, F., D'amico, G., Nair, S., Biondi, A., & Buttazzo, G. (2023). On the real-world adversarial robustness of real-time semantic segmentation models for autonomous driving. IEEE Transactions on Neural Networks and Learning Systems, 35(12), 18328-18342. https://doi.org/10.1109/TNNLS.2023.3314512

[17] Saeedizadeh, N., Jalali, S. M. J., Khan, B., & Mohamed, S. (2025). Cutting-Edge Deep Learning Methods for Image-Based Object Detection in Autonomous Driving: In-Depth Survey. Expert Systems, 42(4), e70020. https://doi.org/10.1111/exsy.70020

[18] Zhang, X., Zhang, Y., Li, Z., Song, Y., Chen, S., Mao, Z., ... & Nie, L. (2025). A real-time cell image segmentation method based on multi-scale feature fusion. Bioengineering, 12(8), 843. https://doi.org/10.3390/bioengineering12080843

[19] Schwonberg, M., Niemeijer, J., Termöhlen, J. A., Schmidt, N. M., Gottschalk, H., & Fingscheidt, T. (2023). Survey on unsupervised domain adaptation for semantic segmentation for visual perception in automated driving. IEEE Access, 11, 54296-54336. https://doi.org/10.1109/ACCESS.2023.3277785

[20] Liu, D., Zhang, D., Wang, L., & Wang, J. (2023). Semantic segmentation of autonomous driving scenes based on multi-scale adaptive attention mechanism. Frontiers in neuroscience, 17, 1291674. https://doi.org/10.3389/fnins.2023.1291674

[21] Plastiras, G., Siddiqui, S., Kyrkou, C., & Theocharides, T. (2020, August). Efficient embedded deep neural-network-based object detection via joint quantization and tiling. In 2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS) (pp. 6-10). IEEE. https://doi.org/10.1109/AICAS48895.2020.9073885

[22] Zhou, K., Liu, Z., Qiao, Y., Xiang, T., & Loy, C. C. (2022). Domain generalization: A survey. IEEE transactions on pattern analysis and machine intelligence, 45(4), 4396-4415. https://doi.org/10.1109/TPAMI.2022.3195549

[23] Zaharia, C., Popescu, V., & Sandu, F. (2023). Hardware–Software Partitioning for Real-Time Object Detection Using Dynamic Parameter Optimization. Sensors, 23(10), 4894. https://doi.org/10.3390/s23104894

[24] Ruhland, J. B., Masoudian, I., & Heider, D. (2025). Enhancing deep neural network training through learnable adaptive normalization. Knowledge-Based Systems, 326, 113968. https://doi.org/10.1016/j.knosys.2025.113968

[25] Wang, R., Yang, T., Liang, C., Wang, M., & Ci, Y. (2025). Reliable autonomous driving environment perception: uncertainty quantification of semantic segmentation. Journal of Transportation Engineering, Part A: Systems, 151(3), 04024117. https://doi.org/10.1061/JTEPBS.TEENG-8660

[26] Yang, S., Wang, W., Liu, C., & Deng, W. (2018). Scene understanding in deep learning-based end-to-end controllers for autonomous vehicles. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 49(1), 53-63. https://doi.org/10.1109/TSMC.2018.2868372

[27] Yang, M., Ewe, L. S., Yew, W. K., Deng, S., & Tiong, S. K. (2025). A Survey of Data Augmentation Techniques for Traffic Visual Elements. Sensors, 25(21), 6672. https://doi.org/10.3390/s25216672

[28] Yuan, X., Li, H., Ota, K., & Dong, M. (2023). Building energy efficient semantic segmentation in intelligent edge computing. IEEE Transactions on Green Communications and Networking, 8(1), 572-582. https://doi.org/10.1109/TGCN.2023.3321113

[29] Ren, W., Tang, Y., Sun, Q., Zhao, C., & Han, Q. L. (2023). Visual semantic segmentation based on few/zero-shot learning: An overview. IEEE/CAA Journal of Automatica Sinica, 11(5), 1106-1126. https://doi.org/10.1109/JAS.2023.123207