2025 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
October 5-8, 2025. Vienna, Austria

2025 IEEE International Conference on Systems, Man, and Cybernetics (SMC) | 979-8-3315-3358-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/SMC58881.2025.11342808

EDRSNet: A Dual-Branch Real-Time Semantic Segmentation Network for UAV Autonomous Flight

Yuanxu Zhu[1,2], Tianze Zhang[2,3], Aiying Wu[1,2], Yang Deng[1,2] and Gang Shi[1,2]

***Abstract*— Autonomous UAV navigation in road scenes is one of the key research focuses in the field. However, vision-based UAV control remains a significant challenge. To address this issue, this paper proposes a real-time semantic segmentation network, EDRSNet, Edge Information Assisted Dual-Branch Encoder Real-Time Semantic Segmentation Net, as the visual perception module for UAVs. On this basis, we introduce the RENA, Ray-based Eight Neighborhood Algorithm, to enhance road extraction and accurately determine the UAV's target point. Combined with a PID control algorithm, this approach enables precise flight control of the UAV. Experimental results show that EDRSNet achieves 51.61 FPS and 17.39 GFLOPs on the DeepGlobe Road dataset and the DRS Road dataset. Furthermore, the integration of EDRSNet with RENA and PID achieves promising results in a simulation environment, with the UAV flight trajectory closely matching the expected path, demonstrating the feasibility of the proposed method for autonomous UAV flight tasks.**

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs), due to their high flexibility and low cost, have demonstrated broad application potential in areas such as traffic monitoring, urban planning, and emergency rescue. Among the many UAV tasks, road extraction from aerial imagery serves as a crucial prerequisite for enabling autonomous navigation and decision-making. However, achieving a balance between accuracy and efficiency under the constraints of limited onboard computational resources remains a core challenge in current research.

In recent years, lightweight semantic segmentation networks such as ENet[1], Fast-SCNN[2], and BiSeNet[3] have been widely proposed to meet real-time requirements. These methods typically adopt an encoder-decoder architecture and incorporate strategies such as dilated convolutions[4], attention mechanisms[5], or multi-scale feature fusion[6] to reduce computational complexity while retaining as much spatial information as possible. However, most of these approaches are designed for ground-level perspectives or static scenes, and their generalization performance significantly degrades in low-altitude aerial views due to issues like scale variation, occlusion, and illumination changes.

This Work was funded by the Key R&D projects of Xinjiang Uygur Autonomous Region, grant number 2022B01006.

[1]School of Computer Science and Technology, Xinjiang University, Urumqi, Xinjiang Uygur Autonomous Region, 830046, China

[2]Xinjiang Key Laboratory of Signal Detection and Processing, Xinjiang University, Urumqi, 830046, China

[3]Faculty of Science,The University of Melbourne, Grattan Street, Parkville, VIC 3010, Australia

*Corresponding author:Gang Shi.(e-mail:shigang@xju.edu.cn)

Yuanxu Zhu and Tianze Zhang have contributed equally to this work. (e-mail:zhuyxv@163.com, zhangtianze.unimelb@gmail.com)

Meanwhile, the field of visual navigation has introduced various flight control strategies, including ORB-SLAM-based localization and mapping[7], deep learning-based path planning algorithms[8], and multi-sensor fusion techniques[9]. While these approaches have achieved certain successes in specific scenarios, most treat perception and control as separate modules, which limits the exploitation of high-level semantic information—such as road topology—and often leads to slow responses or trajectory deviations in complex environments.

To address the above issues, this paper proposes a real-time road extraction model and autonomous flight method that deeply integrates prior-guided semantic segmentation with flight control strategies. On one hand, a lightweight segmentation model that fuses image priors and spatial contextual information is developed to improve road extraction accuracy in complex aerial scenarios. On the other hand, a flight control mechanism based on segmentation results is designed to realize an end-to-end closed-loop flight strategy from semantic perception to path planning and motion control.

The main contributions of this work are as follows:

(1) We propose a novel dual-branch real-time road extraction network—Edge Information Assisted Dual-Branch Encoder Real-Time Semantic Segmentation Net, EDRSNet, which incorporates an edge-enhancement branch in parallel with the semantic segmentation backbone. This design improves both recognition accuracy and processing speed.

(2) We design a lightweight Multi-scale Downsampling Module and a Long-Distance Attention Pyramid Pooling Module. The former adopts cross-stage depthwise separable convolutions, and the latter utilizes a feature pyramid structure to reduce parameters while maintaining feature representation capability, thereby enhancing inference speed and the accuracy of continuous road recognition in complex scenes.

(3) We develop an autonomous flight control strategy based on semantic segmentation results, enabling semantic perception to effectively guide path planning and control. Extensive experiments on public datasets and real UAV platforms demonstrate the proposed method's advantages in terms of generalization, real-time performance, and adaptability to complex environments.

II. RELATIVE WORK

With the development of deep learning, researchers have extensively explored strategies such as multi-scale feature fusion, attention-guided learning, spatial context modeling,

and topological connectivity preservation to improve the accuracy and robustness of road extraction.

In terms of multi-scale feature extraction, Ren et al.[10] proposed the Capsule U-Net, which integrates capsule networks and attention mechanisms to effectively extract multi-scale semantic features and enhance modeling of spatial hierarchies. Li et al.[11] designed a cascaded attention mechanism based on the DenseUNet architecture, combining global and local attention modules to significantly improve the understanding of road contextual structures and enhance feature propagation in dense connections. Tan et al.[12] introduced BSIRNet, which models spatial dependencies through a bidirectional spatial reasoning mechanism, reinforcing semantic consistency in road regions and demonstrating strong generalization capability for large-scale road networks.

For feature modeling in road imagery, Li et al.[13] constructed a multi-level feature sharing architecture combined with directional feature modeling, effectively enhancing the preservation of road topology. Mei et al.[14] proposed the Connectivity-Aware Network (CoANet), which utilizes multi-directional stripe convolutions to capture road orientation and introduces a connectivity cube loss to significantly reduce fragmentation and omissions, thereby improving the overall consistency of road extraction.

Chen et al.[15] combined the strengths of Swin Transformer and ResNet, constructing a dual-branch encoder architecture and incorporating a context-guided filtering module into the skip connections to retain fine details and improve robustness. Zhang et al.[16] proposed a multi-scale representation method that integrates coarse- and fine-grained features. By incorporating a Transformer into the feature fusion module, the model achieves effective cross-scale information interaction. However, despite their strong representational power, Transformer-based models are often limited by high computational cost and inference latency, making them less suitable for resource-constrained platforms.

To meet the demands of edge computing and real-time applications, recent studies have increasingly focused on the design of lightweight road extraction models. Liu et al.[17] proposed LDANet, a lightweight dynamic attention network that combines asymmetric convolution and depthwise separable convolution, balancing low-level feature representation and overall computational efficiency. Qu et al.[18] introduced Road-MobileFormer as a lightweight backbone, incorporating coordinate attention and a micro-label pyramid module to enhance feature localization and reduce model parameters, making it suitable for edge deployment. While these methods offer clear advantages under computational constraints, they still lag behind more complex models in segmentation accuracy.

Vision-based UAV navigation methods have gained wide adoption due to the lightweight nature, low cost, and strong anti-interference of vision sensors. With the advancement of deep learning, accurate autonomous flight control using monocular vision sensors has become feasible. Early visual navigation methods primarily relied on optical flow sensors. Gageik et al.[19] proposed a UAV navigation method that combines optical flow, inertial, ultrasonic, and infrared sensors, using the optical flow sensor for 2D localization, while other sensors aided in obstacle avoidance and error correction—enabling UAV autonomous flight without external reference systems.

With the rapid progress in deep learning, the focus has shifted toward vision-based deep learning approaches for autonomous UAV flight. Arshad et al.[20] proposed a data-driven strategy for UAV navigation in complex dynamic environments based on deep convolutional neural networks, enabling effective autonomous flight. Zhao et al.[21] proposed a deep learning-based autonomous UAV exploration method called DLAE (Deep Learning-based Autonomous UAV Exploration), which integrates position and yaw actions to overcome field-of-view limitations and employs an autoregressive network model to improve exploration efficiency and decision-making speed.

However, most of the aforementioned methods are primarily designed for indoor obstacle avoidance scenarios and are not well-suited for outdoor UAV autonomous flight and remote sensing data acquisition tasks.

III. Method

A. EDRSNet: Edge Information Assisted Dual-Branch Encoder Real-Time Semantic Segmentation Net

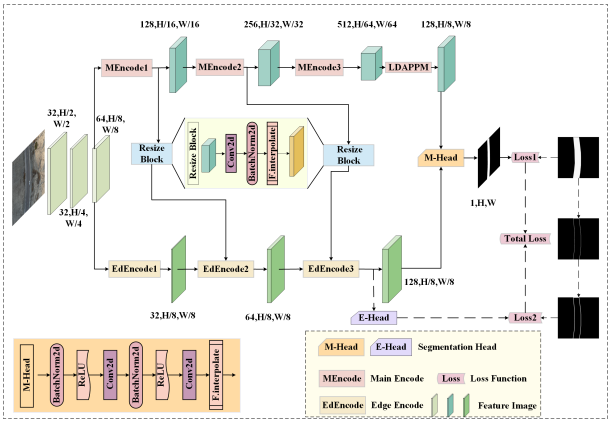

Fig. 1. Overview of EDRSNet for semantic segmentation. The Resize Block is used to share features from the main branch to the edge branch, and its network structure is illustrated in the center of the figure. The structure of M-Head is shown in the lower-left corner. Solid black lines indicate paths where feature processing is applied, while dashed black lines represent paths without feature processing.

The proposed EDRSNet consists of three core modules: a Multi-scale Downsampling Module (MDM), a Dual-Branch Module (DBM), and a Low-Channel Deep Aggregation Pyramid Pooling Module (LDAPPM), as illustrated in Fig. 1. EDRSNet adopts a dual-branch encoder architecture, where the two branches are designed to separately extract deep contextual features and edge features. Semantic and edge features are processed in parallel, and the decoder is removed to reduce convolutional operations. Instead, a lightweight segmentation head is used for final prediction, thereby significantly reducing the model's computational cost.

In the EDRSNet framework, the input image (H×W×3) is first passed through the MDM, which downsamples the spatial resolution to H/8 × W/8 while increasing the feature dimension to 64. To enhance feature representation and gradient flow, residual connection blocks are interleaved throughout the downsampling process. The MDM effectively expands the receptive field through downsampling and convolution, and the reduced resolution of the output features significantly lowers the input size to the subsequent dual-branch encoder, thereby reducing FLOPs and computational complexity and improving real-time performance.

The output feature from MDM (H/8 × W/8 × 64) is fed into both the main semantic branch and the edge branch for dual-branch feature extraction. The main branch focuses on aggregating local and global contextual information to capture long-range dependencies. It adopts a hierarchical progressive encoder structure, enhanced with residual and bottleneck blocks, to improve feature expressiveness. After processing, the resolution is further reduced to H/64 × W/64 and the channel dimension is expanded to 512.

In parallel, the edge branch focuses on extracting high-frequency features and predicting road boundaries. It uses multi-level feature fusion and edge refinement operators to enhance boundary detection. The input (H/8 × W/8 × 64) maintains its spatial resolution, while the channel dimension is first compressed to 32 and then gradually expanded to 128.

To improve feature representation, feature fusion is applied from the main branch to the edge branch. Specifically, the resolution of the main branch features is upsampled back to H/8 × W/8 and fused with the edge features, enriching them with additional contextual information.

Finally, the LDAPPM module performs progressive fusion of multi-scale pooled features using cascaded 3×3 convolutions. This produces a deep information stream that enhances the integration of semantic and spatial details. The resulting feature map (H/8 × W/8 × 128) is then fused with the edge branch output through element-wise addition and fed into the M-Head segmentation head for final prediction.

B. Multi-scale Downsampling Module

The Multi-scale Downsampling Module (MDM) progressively downsamples input features to create lower-resolution maps, significantly reducing computational complexity by processing fewer pixels in deeper layers. This improves runtime efficiency, particularly in deep networks where feature map sizes rapidly increase. Downsampling helps alleviate this issue.

In the MDM, each downsampling operation uses a convolution with stride 2 to halve the spatial resolution. Inside each Basic Block, convolutions with stride 1 focus on the channel dimension for better local feature extraction. Residual connections add the input feature to the output, preserving gradient flow to shallow layers and enhancing model trainability and stability.

To prevent the loss of spatial and semantic information, and to address gradient vanishing, Basic Block units are incorporated during downsampling. These units, consisting of multiple convolutions and residual connections, refine features and improve model stability during training. The structure of the MDM is shown in Fig. 2.

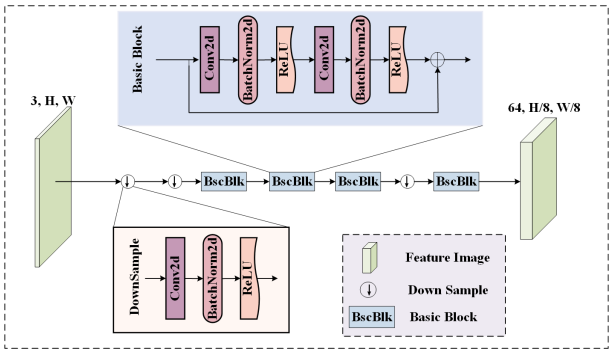

Fig. 2. Structure of the Multi-scale Downsampling Module. The Basic Block is a residual block, and its structure is illustrated at the top of the figure. The downsampling structure is shown at the bottom.

C. Double Branch Module

The Double Branch Module consists of three components: a main branch, an edge auxiliary branch, and a multi-task joint supervision module. The main and edge branches are responsible for extracting deep semantic features and edge features, respectively.

The main branch is designed to aggregate both local and global contextual information, enabling the modeling of long-range dependencies. It adopts a hierarchical progressive encoder architecture, which leverages multi-stage feature extraction and contextual aggregation mechanisms to facilitate semantic understanding and long-distance interaction modeling, as shown in Fig. 3. The main branch takes the output of the Multi-scale Downsampling Module (MDM) as input and is composed of a series of Downsampling, Basic Blocks, and Bottleneck Blocks.

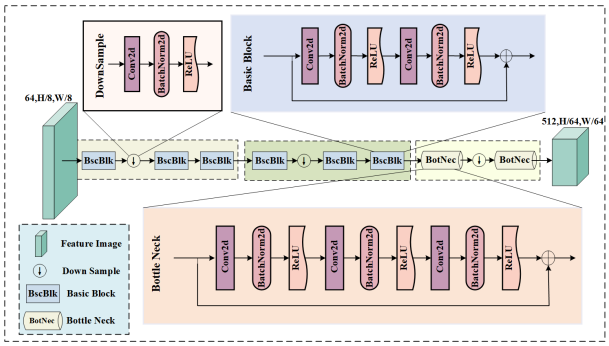

Fig. 3. Structure of the main branch. The Bottle Neck module is a bottleneck block, and its structure is illustrated at the bottom of the figure.

The first Basic Block in the encoder increases the number of channels to enhance feature representation. Subsequent blocks maintain resolution and channel size, using iterative convolutions with local receptive fields to strengthen neighborhood feature correlation, enabling fine-grained texture and shape extraction for accurate localization. Downsampling

layers employ 1×1 convolutions with stride 2 to reduce spatial resolution, compressing redundant data while preserving key features and improving efficiency.

Bottleneck blocks are introduced as the network deepens, balancing efficiency and feature representation. The "expand–squeeze–expand" design focuses on discriminative channels, while deep 3×3 convolutions enlarge the receptive field to model long-range dependencies. This modular stacking allows the main branch to capture both local sensitivity and global consistency for detailed semantic representations.

The edge branch uses the Laplacian operator for edge detection, enhancing high-frequency responses to capture sharper gradients. A dilation operation is applied to the Gaussian-blurred Laplacian results to improve edge clarity. The dilated edge label is computed as:

$$E_{\text{dilate}}(x, y) = \bigcup_{(i,j) \in S} E(x + i, y + j) \tag{1}$$

The results are visualized in Fig. 4.

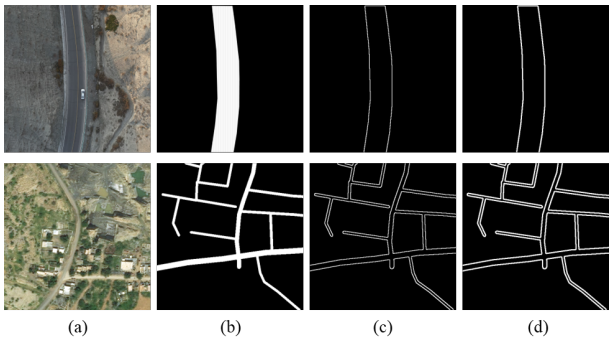

Fig. 4. Comparison of edge detection results using Laplacian operator: (a) Original image, (b) Label, (c) Laplacian edge map, (d) Dilated edge map.

The edge branch focuses on extracting high-frequency information and predicting road boundaries. It employs multi-level feature fusion and edge-enhancing operators to construct a hierarchical edge-aware network, as illustrated in Fig. 5. After compressing spatial redundancy through downsampling, the input features are fused with those from the main branch to generate enriched feature maps. These features are further refined using alternating Basic Blocks and Bottleneck Blocks, progressively increasing the number of channels to 128 while maintaining the spatial resolution to preserve detailed edge information.

An edge segmentation head, composed of a 1×1 convolution followed by bilinear upsampling, maps the 128-channel features to a probability map. The predicted edge map is supervised by the dilated Laplacian edge label using a weighted cross-entropy loss. The edge branch is trained in parallel with the main branch, and its outputs guide the main features via an attention mechanism, enhancing the response strength at road boundary regions and improving segmentation accuracy.

The multi-task joint supervision strategy imposes dual loss constraints. It computes loss between the predicted

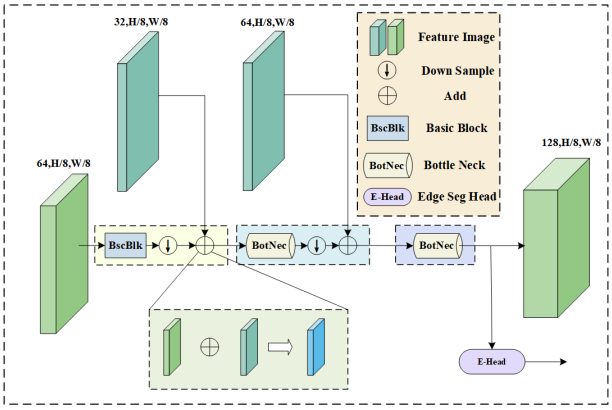

Fig. 5. Structure of the edge auxiliary branch

segmentation map and the ground truth, as well as between the edge branch output and the Laplacian-based edge labels. The total loss is defined as:

$$Loss = \lambda_1 loss_1 + \lambda_2 loss_2 \tag{2}$$

Where $\lambda_1 = 1$ and $\lambda_2 = 9$. $loss_1$ is the combined Dice loss and Focal loss used to supervise overall segmentation accuracy, $loss_2$ applies the same loss functions to the edge branch prediction and the dilated Laplacian label to enhance learning of difficult edge samples.

Through joint optimization of semantic segmentation and edge detection, the strategy enhances the model's performance. In early stages, both branches share shallow convolutional features, enabling the edge branch to leverage textural features learned by the main branch. This sharing preserves feature integrity and improves segmentation precision and overall performance.

D. Low-Channel Deep Aggregation Pyramid Pooling Module

To better model long-range contextual dependencies and effectively fuse features from the edge branch, we propose the Low-Channel Deep Aggregation Pyramid Pooling Module (LDAPPM). This module captures global contextual information using large-kernel pooling, enhancing the network's perception of global semantics. Additionally, it employs a progressive feature aggregation strategy to integrate multi-scale context information, improving the fusion of semantic and detailed features. The overall structure of LDAPPM is illustrated in Fig. 6.

The input to LDAPPM is a low-resolution feature map of size (512, H/64, W/64). Using such a small-scale input significantly reduces the parameter count and computational cost of subsequent convolution and pooling operations, thereby improving inference efficiency. The input feature map is processed by a set of multi-level pooling blocks, producing feature maps of the same number of channels but at different spatial resolutions. These multi-scale features are aggregated using a progressive fusion path: each pooling result is fused step-by-step via cascaded 3×3 convolutions.

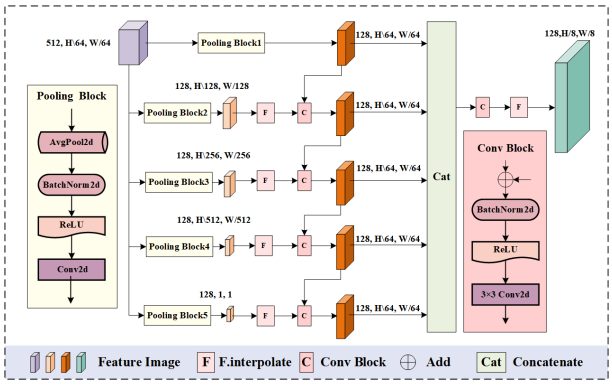

Fig. 6. Structure of the LDAPPM module. The Pooling Block, used for multi-scale pooling operations, is illustrated on the left side of the figure. The C module represents a convolution block, shown on the right side.

The output from each stage is first fused with the feature map at the $1/64$ resolution, and then progressively merged with coarser-level features (e.g., $1/256$, $1/512$), forming a deep information stream that enhances the integration of semantic context and spatial detail. At each level, the current feature map is resized via interpolation and fused with the output from the previous stage using an element-wise addition (ADD) operation, maintaining the feature size at (128, H/64, W/64). After the fusion of all pooling levels, the resulting features are concatenated along the channel dimension and compressed, avoiding channel explosion that can occur with naive multi-scale concatenation. Finally, a convolution followed by bilinear upsampling is applied to produce the output feature map of size (128, H/8, W/8).

E. UAV Control

The core of autonomous UAV flight lies in achieving accurate flight control, enabling the UAV to dynamically adjust according to environmental information and reliably accomplish its tasks. To this end, this paper first addresses the problem of target point acquisition by proposing an improved Ray-based Eight Neighborhood Algorithm (RENA) [31]. Traditional eight-neighborhood algorithms suffer from issues such as edge discontinuity and high computational complexity. RENA resolves these by optimizing the breadth-first search strategy into a directionally constrained search, reducing the time complexity to $O(n)$.

In computation, RENA starts by generating rays from the midpoint of the bottom edge of the image, scanning in four directions ($\pm 30^\circ$ and $\pm 60^\circ$). When a transition from white to black is detected, the point is marked as a boundary point and used as the starting node. This method yields four initial points. Then, a direction-constrained search is performed: the left and right boundaries of the road are searched using counterclockwise and clockwise directions, respectively. When another white-to-black transition is detected, the search pauses, and the transition point becomes the new start for the next iteration. This process continues until the full road boundaries are extracted, as shown in Fig. 7. To mitigate misrecognition, the algorithm compares the number of detected points on both sides and selects the one with more points as the valid boundary.

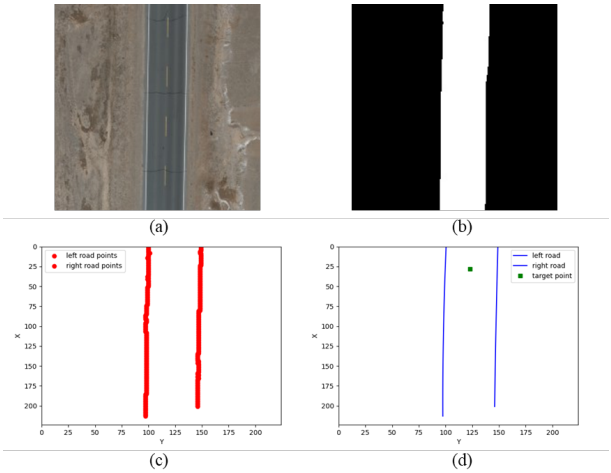

Fig. 7. RENA processing results: (a) Original image, (b) Predicted segmentation map from EDRSNet, (c) Extracted boundaries by RENA, (d) Fitted road boundaries and computed target point.

The complete UAV control pipeline is shown in Fig. 8, consisting of three stages: data acquisition, visual processing, and flight control.

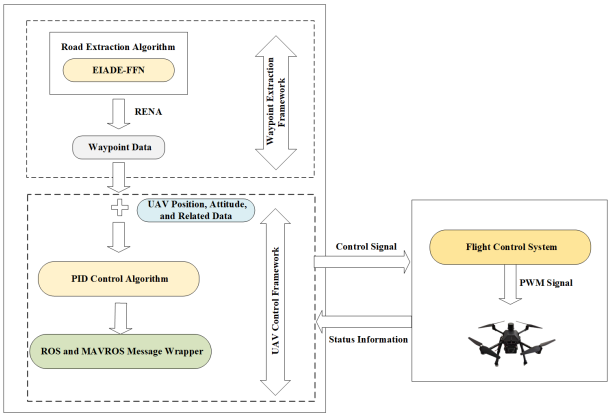

Fig. 8. UAV Flight Control Framework. The UAV flight control system mainly consists of three components. The semantic segmentation model and the RENA algorithm form the Waypoint Extraction module, which extracts coordinate point information and sends it to the Control Command Generation module. This module utilizes a PID algorithm to compute velocity and yaw angle commands. These commands are then transmitted via MAVROS to the flight control system (FCS) of the UAV, which generates control signals to guide the UAV's flight.

In the data acquisition stage, road images are typically captured by visual sensors. However, in this study, since simulation environments are used, ROS is employed to publish road video nodes instead.

During the visual processing stage, ROS subscribes to the road video node, and the video stream is fed into EDRSNet to generate road segmentation maps. The RENA algorithm is then applied to the predicted maps to extract geometrically consistent road boundaries, suppress false detections, and compute the target point coordinates $(x_{\text{target}}, y_{\text{target}})$.

In the control stage, a PID controller receives the target waypoint $(x_{\text{target}}, y_{\text{target}})$ from the RENA module and the UAV's current pose $(x_{\text{current}}, y_{\text{current}}, \theta)$. It then computes position and heading control. The position control relies on the lateral error $e_x = x_{\text{target}} - x_{\text{current}}$ and longitudinal error $e_y = y_{\text{target}} - y_{\text{current}}$, and generates velocity commands via 2D PID control, as in:

$$v_x = K_{p,x} e_x + K_{i,x} \left(\sum e_x \Delta t \right) + K_{d,x} \frac{de_x}{dt} \tag{3}$$

$$v_y = K_{p,y} e_y + K_{i,y} \left(\sum e_y \Delta t \right) + K_{d,y} \frac{de_y}{dt} \tag{4}$$

Here, Δt is the MAVROS message update interval (set to 25 ms), and the integral term is computed using a sliding window to prevent integral wind-up.

Heading control is determined by the yaw error $e_\theta = \arctan 2(e_y, e_x) - \theta$, and its control output is computed via a separate PID controller as in:

$$u_\theta = K_{p,\theta} e_\theta + K_{i,\theta} \sum e_\theta \Delta t + K_{d,\theta} \frac{de_\theta}{dt} \tag{5}$$

Once v_x, v_y, and u_θ are calculated, the commands are packaged into MAVROS-compliant messages and transmitted to the PX4 flight controller. The controller's internal mixer then adjusts rotor thrust in real time to execute the desired flight trajectory.

IV. EXPERIMENTS

A. Comparative Experiments of EDRSNet

In this study, the DeepGlobe Road and the DRS Road dataset[22] are selected for training and testing. The Adam optimizer is used for parameter optimization, with an initial learning rate set to 2e-4. To prevent overfitting, Dropout regularization with a dropout rate of 0.2 is applied. During training, if the loss no longer decreases, the learning rate is progressively reduced by a factor of 0.2. Training is terminated when the learning rate falls below 5e-7, ensuring optimal training effectiveness and model performance. The overall loss function is a combination of the Dice loss and Focal loss. All experiments were conducted on a workstation equipped with an NVIDIA Tesla A40 GPU.

The results of the quantitative comparison are shown in the Table I and II.

It can be observed that the proposed EDRSNet model achieves the best performance in terms of FLOPs and FPS. Compared to Mobile-Seed, it improves IoU and F1-score by 0.85% and 0.48%, respectively. Although there is a slight decrease in IoU and F1-score compared to PIDNet-L, the FPS is significantly higher, demonstrating that the model achieves a good balance between real-time performance and segmentation accuracy.

The visualization result of the model is shown in Figure 9.

TABLE I

QUANTITATIVE COMPARISON RESULTS OF DIFFERENT MODELS ON THE DEEPGLOBE ROAD DATASET

Methods	IoU	F1-Score	Params (M)	FLOPs (G)	FPS
LHU-Net[23]	63.22%	77.46%	10.52	51.75	22.74
PIDNet-L[24]	64.36%	78.32%	36.97	137.92	30.58
Mobile-Seed[25]	63.57%	77.72%	2.45	19.47	49.57
SegFormer-B0[26]	62.97%	77.28%	3.82	62.65	27.62
CoANet[14]	64.97%	78.75%	59.14	277.41	0.45
BiSeNetV2[27]	**64.98%**	**78.77%**	3.61	12.91	3.75
STDC1[28]	59.18%	74.36%	14.23	23.52	4.72
Ulite[29]	62.15%	76.66%	**0.87**	**6.13**	4.83
HyperSeg-S[30]	63.72%	77.84%	10.26	18.03	45.13
SFNet[31]	62.88%	77.21%	12.87	123.48	18.85
EDRSNet [ours]	63.32%	77.54%	7.56	17.39	**51.61**

TABLE II

QUANTITATIVE COMPARISON RESULTS OF DIFFERENT MODELS ON THE DRS ROAD DATASET

Methods	IoU	F1-Score	Params (M)	FLOPs (G)	FPS
LHU-Net	84.45%	91.57%	10.52	51.75	22.74
PIDNet-L	87.24%	93.18%	36.97	137.92	30.58
Mobile-Seed	86.02%	92.48%	2.45	19.47	49.57
SegFormer-B0	84.30%	91.48%	3.82	62.65	27.62
CoANet	**89.05%**	**94.21%**	59.14	277.41	0.45
BiSeNetV2	88.01%	93.62%	3.61	12.91	3.75
STDC1	86.90%	92.98%	14.23	23.52	4.72
Ulite	87.11%	93.11%	**0.87**	**6.13**	4.83
HyperSeg-S	86.56%	92.79%	10.26	18.03	45.13
SFNet	83.70%	91.13%	12.87	123.48	18.85
EDRSNet [ours]	86.85%	92.96%	7.56	17.39	**51.61**

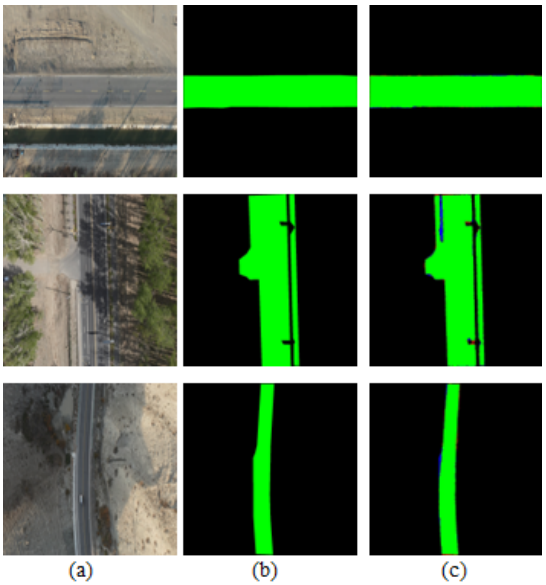

Fig. 9. Results of EDRSNet on the DRS Road dataset. (a) Original image, (b) Ground truth label, (c) Prediction generated by EDRSNet.

B. Ablation Study of EDRSNet

The EDRSNet model primarily consists of three core modules: the Multi-scale Downsampling Module, the Double Branch Module, and Low-Channel Deep Aggregation Pyramid Pooling Module. To assess the individual contribution of each component, ablation studies are conducted based on these modules. Since the architecture must retain at least one encoder branch, the DBM evaluation is divided into configurations with and without the edge branch. In the absence of the LDAPPM module, feature maps from the encoder are upsampled using interpolation.

TABLE III
ABLATION STUDY RESULTS OF EDRSNET ON THE DEEPGLOBE ROAD DATASET

No.	A	B	C	Accuracy	IoU	Precision	Recall	F1-Score
1				98.03%	61.41%	77.34%	74.88%	76.09%
2	✓			98.04%	61.65%	77.53%	75.07%	76.28%
3	✓	✓		98.09%	62.52%	78.01%	75.89%	76.94%
4	✓		✓	98.08%	62.34%	77.99%	75.65%	76.80%
5	✓	✓	✓	**98.14%**	**63.32%**	**78.55%**	**76.56%**	**77.54%**

Note: A is MDM, B is DBM, and C is LDAPPM.

- **No.1**: Baseline model with consecutive downsampling in the early feature processing stage, without residual blocks for feature extraction;
- **No.2**: Builds on No.1 by introducing the MDM module for enhanced downsampling; only the main encoder branch is used, with upsampling to restore feature map resolution;
- **No.3**: Extends No.2 by adding the edge branch and multi-task supervision, with feature fusion between the main and edge branches to form the DBM;
- **No.4**: Extends No.2 by replacing upsampling with the LDAPPM module;
- **No.5**: Combines all three modules (MDM, DBM, LDAPPM) into the full EDRSNet architecture;

Compared to No.1, No.2 improves IoU, Precision, and Recall by 0.24%, 0.19%, and 0.18%, respectively, showing that the residual connections in MDM enhance feature extraction.

No.3 outperforms No.2 by 0.86%, 0.48%, and 0.82% in IoU, Precision, and Recall, indicating that the edge branch and feature fusion improve boundary awareness and foreground-background distinction.

No.4 improves on No.2 with gains of 0.69%, 0.46%, and 0.58% in IoU, Precision, and Recall, suggesting LDAPPM effectively recovers unrecognized regions. However, compared to No.3, No.4 sees a slight decrease of 0.18% in IoU and 0.24% in Recall, while Precision increases by 0.02%. This suggests DBM better addresses false positives due to edge guidance, while LDAPPM focuses on false negatives.

No.5, the full EDRSNet model, achieves the best performance. Compared to No.3, it improves IoU, Precision, and Recall by 0.80%, 0.53%, and 0.68%, respectively, and compared to No.4, gains are 0.98%, 0.56%, and 0.91%. These results confirm the complementary nature of the modules, improving feature representation and sample discrimination.

C. Evaluation of the RENA

To validate the performance of the proposed RENA algorithm, we conducted comparative experiments under identical hardware conditions, using input images with a resolution of 224 $\times$ 224. The results are shown in Table IV. As shown in Table IV, compared to the traditional eight-neighborhood algorithm, the RENA algorithm significantly reduces computational time, laying a solid foundation for real-time and precise UAV navigation.

TABLE IV
COMPARISON BETWEEN RENA AND TRADITIONAL EIGHT-NEIGHBORHOOD ALGORITHM

Algorithm	TC	SC	Computation Time
8-neighborhood	$O(N^2)$	$O(N^2)$	9200ms
RENA	$O(N)$	$O(N)$	10ms

D. UAV Experiments

The UAV's flight status in the simulation environment is shown in Fig. 10. The three video frames represent the road scene, real-time semantic segmentation result, and the output of the eight-neighborhood algorithm, respectively. The red box indicates the current position of the UAV, while the green box marks the UAV's starting position.

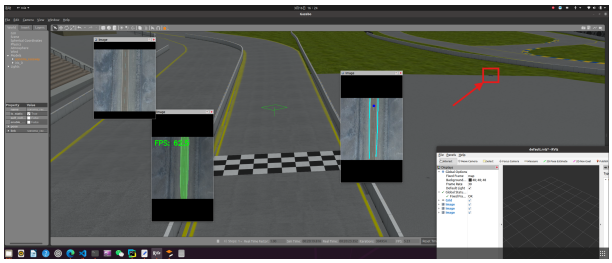

Fig. 10. UAV flight status and video processing results in the simulation environment

The UAV flight trajectory using the proposed model in the simulation environment is shown in Fig. 11. Under the control of EDRSNet, the UAV follows a trajectory that closely matches the desired path. It covers a total distance of 150 meters in 60.73 seconds, resulting in an average flight speed of 2.47 m/s.

V. CONCLUSIONS

We propose EDRSNet, a real-time semantic segmentation network, and an autonomous UAV flight control method, validated in various scenarios. EDRSNet reduces parameters and complexity with compressed features and residual connections, while maintaining feature robustness. Its dual-branch architecture extracts both deep semantics and fine-grained edge details, aiding in road direction understanding. LDAPPM is introduced to fuse multi-scale contextual information, improving feature integration. For UAV control, the RENA algorithm refines segmentation outputs and extracts target points, which are then used in a PID control algorithm

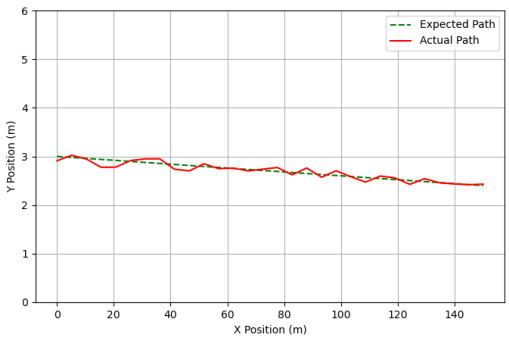

Fig. 11. Flight trajectory of the UAV under the control of EDRSNet

for velocity and yaw commands, transmitted via MAVLINK for autonomous flight. Experimental results show EDRSNet's superior real-time performance, with the UAV control algorithm performing well in simulations.

ACKNOWLEDGMENT

Y.Z. and T.Z. conceptualized the study; Y.Z. , T.Z. and A.W. curated and processed the data; Y.Z. and T.Z. conducted formal analysis; G.S. acquired funding, provided resources, participated in the investigation, and supervised the project; Y.Z. designed the methodology; Y.Z. and T.Z. managed the project's progress and coordination;Y.Z. and T.Z. validated the results and created visualizations; Y.Z. wrote the main manuscript text; Y.Z. , T.Z. and Y.D. reviewed and edited the manuscript.

REFERENCES

[1] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, 'Enet: A deep neural network architecture for real-time semantic segmentation', arXiv preprint arXiv:1606. 02147, 2016.

[2] R. P. K. Poudel, S. Liwicki, and R. Cipolla, 'Fast-scnn: Fast semantic segmentation network', arXiv preprint arXiv:1902. 04502, 2019.

[3] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. S. BiSeNet, 'v2: Bilateral network with guided aggregation for real-time semantic segmentation., 2021, 129', DOI: https://doi. org/10. 1007/s11263-021-01515-2, pp. 3051–3068.

[4] F. Yu, V. Koltun, and T. Funkhouser, 'Dilated residual networks', in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 472–480.

[5] A. Vaswani et al., 'Attention is all you need', Advances in neural information processing systems, vol. 30, 2017.

[6] J. Wang, ÒFundamentals of erbium-doped fiber amplifiers arrays (Periodical styleÑSubmitted for publication),Ó IEEE J. Quantum Electron., submitted for publication.

[7] Q. Fu et al., 'Fast ORB-SLAM without keypoint descriptors', IEEE transactions on image processing, vol. 31, pp. 1433–1446, 2021.

[8] R. Chai, D. Liu, T. Liu, A. Tsourdos, Y. Xia, and S. Chai, 'Deep learning-based trajectory planning and control for autonomous ground vehicle parking maneuver', IEEE Transactions on Automation Science and Engineering, vol. 20, no. 3, pp. 1633–1647, 2022.

[9] M. H. Harun, S. S. Abdullah, M. S. M. Aras, and M. B. Bahar, 'Sensor fusion technology for unmanned autonomous vehicles (UAV): A review of methods and applications', in 2022 IEEE 9th International Conference on Underwater System Technology: Theory and Applications (USYS), 2022, pp. 1–8.

[10] Y. Ren, Y. Yu, and H. Guan, 'DA-CapsUNet: A dual-attention capsule U-Net for road extraction from remote sensing imagery', Remote Sensing, vol. 12, no. 18, p. 2866, 2020.

[11] J. Li, Y. Liu, Y. Zhang, and Y. Zhang, 'Cascaded attention DenseUNet (CADUNet) for road extraction from very-high-resolution images', ISPRS International Journal of Geo-Information, vol. 10, no. 5, p. 329, 2021.

[12] H. Tan, H. Xu, and J. Dai, 'BSIRNet: A road extraction network with bidirectional spatial information reasoning', Journal of Sensors, vol. 2022, no. 1, p. 6391238, 2022.

[13] X. Li, Y. Wang, L. Zhang, S. Liu, J. Mei, and Y. Li, 'Topology-enhanced urban road extraction via a geographic feature-enhanced network', IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 12, pp. 8819–8830, 2020.

[14] J. Mei, R.-J. Li, W. Gao, and M.-M. Cheng, 'CoANet: Connectivity attention network for road extraction from satellite imagery', IEEE Transactions on Image Processing, vol. 30, pp. 8540–8552, 2021.

[15] T. Chen, D. Jiang, and R. Li, 'Swin transformers make strong contextual encoders for VHR image road extraction', in IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium, 2022, pp. 3019–3022.

[16] Z. Zhang, C. Miao, C. Liu, and Q. Tian, 'DCS-TransUperNet: Road segmentation network based on CSwin transformer with dual resolution', Applied Sciences, vol. 12, no. 7, p. 3511, 2022.

[17] B. Liu, J. Ding, J. Zou, J. Wang, and S. Huang, 'LDANet: a lightweight dynamic addition network for rural road extraction from remote sensing images', Remote Sensing, vol. 15, no. 7, p. 1829, 2023.

[18] G. Qu et al., 'Road-MobileSeg: Lightweight and Accurate Road extraction model from Remote sensing images for Mobile devices', Sensors, vol. 24, no. 2, p. 531, 2024.

[19] N. Gageik, M. Strohmeier, and S. Montenegro, 'An autonomous UAV with an optical flow sensor for positioning and navigation', International Journal of Advanced Robotic Systems, vol. 10, no. 10, p. 341, 2013.

[20] M. A. Arshad et al., 'Drone Navigation Using Region and Edge Exploitation-Based Deep CNN', IEEE Access, vol. 10, pp. 95441–95450, 2022.

[21] Y. Zhao, J. Zhang, and C. Zhang, 'Deep-learning based autonomous-exploration for UAV navigation', Knowledge-Based Systems, vol. 297, p. 111925, 2024.

[22] Y. Zhu, T. Zhang, A. Wu, and G. Shi, 'PISCFF-LNet: A Method for Autonomous Flight of UAVs Based on Lightweight Road Extraction', Drones, vol. 9, no. 3, p. 226, 2025.

[23] Y. Sadegheih, A. Bozorgpour, P. Kumari, R. Azad, and D. Merhof, 'Lhu-net: A light hybrid u-net for cost-efficient, high-performance volumetric medical image segmentation', arXiv preprint arXiv:2404. 05102, 2024.

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

[25] Y. Liao et al., 'Mobile-seed: Joint semantic segmentation and boundary detection for mobile robots', IEEE Robotics and Automation Letters, 2024.

[26] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, 'SegFormer: Simple and efficient design for semantic segmentation with transformers', Advances in neural information processing systems, vol. 34, pp. 12077–12090, 2021.

[27] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, 'Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation', International journal of computer vision, vol. 129, pp. 3051–3068, 2021.

[28] M. Fan et al., 'Rethinking bisenet for real-time semantic segmentation', in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 9716–9725.

[29] B.-D. Dinh, T.-T. Nguyen, T.-T. Tran, and V.-T. Pham, '1M parameters are enough? A lightweight CNN-based model for medical image segmentation', in 2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2023, pp. 1279–1284.

[30] Y. Nirkin, L. Wolf, and T. Hassner, 'Hyperseg: Patch-wise hypernetwork for real-time semantic segmentation', in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 4061–4070.

[31] X. Li et al., 'Semantic flow for fast and accurate scene parsing', in Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16, 2020, pp. 775–793.