A Real-time Semantic Elevation Map Construction Method Based on Hierarchical Residual Network

Zhe Ma
College of Information Science and Technology, Beijing University of Chemical Technology
Beijing, China
mz0810@163.com

2025 8th International Conference on Advanced Algorithms and Control Engineering (ICAACE) | 979-8-3315-3508-7/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICAACE65325.2025.11018910

***Abstract*—Semantic elevation map integrates spatial geometric information with semantic attributes, providing multimodal perception and decision-making support for mobile robots. However, existing mapping methods face challenges when handling object-background boundary ambiguities in multi-object dynamic environments, leading to reduced accuracy in defining traversable areas. To address these challenges, we propose a real-time semantic elevation map construction method based on hierarchical residual network: 1) An improved YOLACT segmentation model with hierarchical residual connections is employed to generate image segmentation masks for stairs, pedestrians, and obstacles; 2) The 2D segmentation masks are fused with 3D point cloud features to compute environmental height information and determine semantic attributes, constructing the semantic elevation map. Experimental results demonstrate that the improved image segmentation module outperforms existing methods, and the semantic elevation map accurately estimates both the terrain height and semantic categories of the environment.**

Keywords-semantic segmentation; hierarchical residual network; semantic elevation map

I. INTRODUCTION

Elevation map is an environmental representation method that describes the height of terrain or object surfaces in space. It is characterized by a simple structure and strong scalability [1]. The core principle involves generating a grid-based height representation from data acquired by depth sensors, thereby capturing the topographical variations of the surrounding environment. The elevation map not only enables robots to recognize key vertical structures, such as stairs and walls, but also assists in path planning and motion decision-making by perceiving height differences between regions. This demonstrates its significant potential for robotic navigation tasks.

Currently, elevation map-based robotic navigation methods can be categorized into two main approaches: geometry-based methods and multimodal methods that integrate semantic information. The geometry-based approach is the most commonly used, as it determines terrain geometric features by analyzing height variations on the terrain surface, which subsequently informs path planning for robotic navigation tasks [2]. In related studies, Fankhauser *et al*. [3] propose a probabilistic terrain mapping method based on robot self-localization, which integrates kinematic and inertial measurement data to generate local elevation maps with confidence intervals. This approach mitigates localization uncertainty and enhances the navigation performance of mobile robots in complex terrains. Jenelten *et al*. [4] propose a terrain-aware motion optimization method specifically designed for quadrupedal robots, significantly enhancing footstep optimization and postural stability for multi-legged robots on irregular terrains. Mishra *et al*. [5] develop a terrain mapping method based on planar regions, where kinematic, inertial, and depth sensor data are combined to extract planar surfaces. The resulting elevation map is memory-efficient and computationally fast, making it suitable for dynamic gait planning and navigation tasks in complex environments.

However, geometry-based methods face certain limitations in complex environments. They rely heavily on prior knowledge, which not only raises deployment costs but also restricts broader applicability. Furthermore, since elevation maps represent the environment exclusively based on geometric information, they often struggle to distinguish between structures that appear similar or have minimal height differences, thereby impairing subsequent navigation decisions. Consequently, the practical performance of geometry-based methods in real-world scenarios remains suboptimal.

To overcome these limitations, researchers have actively explored methods to augment elevation maps with multimodal information, aiming to improve downstream task performance. With the rapid development of deep learning, numerous studies have employed neural networks for scene-level object classification and integrated these classification outcomes with elevation maps to incorporate semantic information. This augmented representation is commonly referred to as a Semantic Elevation Map (SEM). Maturana *et al*. [6] propose an improved semantic mapping system that integrates geometric and semantic information, effectively distinguishing different terrain types—such as paths, grass, and obstacles—in outdoor off-road environments. This system also enables dynamic updates to grid maps, significantly improving the adaptability and robustness of path planning in off-road navigation. Ewen *et al*. [7] introduce a Bayesian inference framework for real-time terrain feature estimation. By leveraging RGB-D camera data, their approach performs recursive probabilistic estimation of terrain surface contours and feature distributions, overcoming the limitations of traditional methods. Erni *et al*. [8] develop a multimodal elevation map framework that fuses data from multiple sensors, including point clouds, RGB images, and semantic information. This approach provides a flexible and efficient environmental representation, allowing robots to better understand and navigate complex environments while optimizing path planning based on different terrain types.

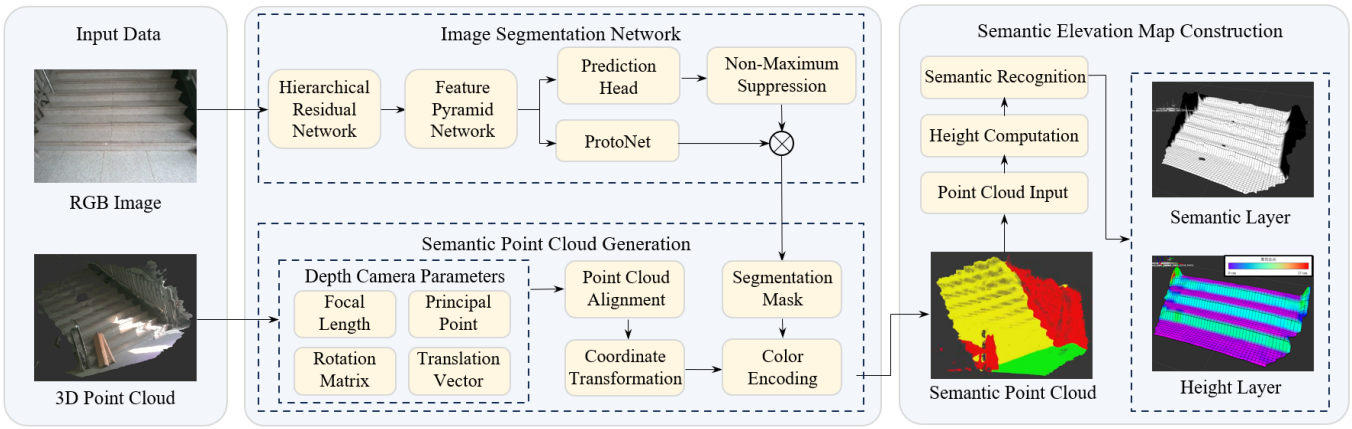


Figure 1. Algorithm Process.

Existing research on SEM primarily focuses on enhancing geometric mapping accuracy and information fusion efficiency, while often overlooking the improvement of semantic accuracy in the map. In dynamic, multi-object environments, factors such as inter-object occlusions reduce the precision of object-background boundary delineation, leading to errors in identifying traversable paths. Consequently, these inaccuracies pose potential risks in path planning and navigation.

In summary, SEM enhances a robot's capacity to understand its environment. However, existing approaches still face limitations, such as insufficient adaptability to dynamic settings, high computational overhead, and difficulties in accurately identifying critical environmental features. To address these challenges, this paper proposes a real-time SEM construction method based on a hierarchical residual network. Specifically, a depth camera is used to capture both RGB images and point cloud data. An improved YOLACT network, incorporating hierarchical residual connections, then generates segmentation masks. These masks are subsequently aligned and fused with the point cloud to produce a semantic point cloud, from which environmental height information is extracted and semantic attributes are determined to build the SEM. Figure 1 provides an overview of the proposed algorithm's workflow.

II. Proposed Methods

A. Image Segmentation

Constructing a SEM first requires determining the semantic categories of different regions in the environment. YOLACT (You Only Look at Coefficients) [9] is an instance segmentation model based on convolutional neural networks. It efficiently performs image segmentation by predicting prototype masks and mask coefficients in parallel. However, due to the fixed receptive field of conventional convolutional units, the YOLACT model struggles to simultaneously capture global semantic information and precise local edge features. To ensure a lightweight network while expanding the receptive field of deep features, we incorporate hierarchical residual connections [13] into the traditional convolutional units. The constructed Conv2d* is shown in Figure 2(b), the improved convolutional unit divides the input feature map into k subsets, denoted as Q_i, where i ∈{0, 1, 2, …, k-1}. Each subset retains the same spatial

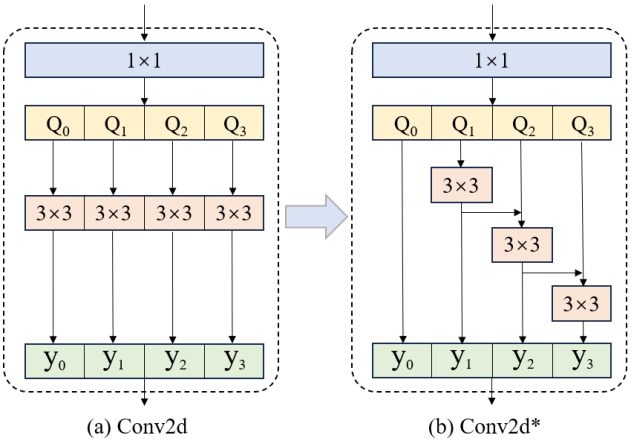


Figure 2. Comparison of improved convolutional units.

resolution as the input feature map but reduces the number of channels to 1/k of the original. Except for Q_0, each Q_i is processed by a 3×3 convolution kernel, denoted as $f_i()$, producing the output feature map y_i. The final output of the convolutional unit is $\sum_{i=0}^{k-1} y_i$, where Σ represents the feature concatenation operation. The formulation is given as follows:

$$y_i = \begin{cases} Q_i & i = 0 \\ f_i(Q_i) & i = 1 \\ f_i(Q_i + y_{i-1}) & 1 < i \le k \end{cases} \quad (1)$$

By improving the convolutional unit, feature maps are processed at multiple scales, enhancing the expressiveness of deep features while reducing redundant computations through parameter reduction. Building on Conv2d*, we further improve the YOLACT network by replacing traditional convolutional units in its backbone with Conv2d*. This modification enables efficient feature propagation across different hierarchical levels within the deep network, mitigating issues such as gradient vanishing and information loss, which commonly affect deep architectures. The improved backbone network maintains a lightweight structure while achieving a gradual expansion of the receptive field. This allows the network to not only precisely capture the edge details of small objects but also effectively distinguish background regions in complex scenes, thereby enhancing the accuracy of semantic segmentation.

B. Semantic Elevation Map Construction

1) Semantic Point Cloud

After obtaining the segmentation masks, it is necessary to spatially align and fuse the point cloud data with the masks to generate a semantic point cloud. First, the intrinsic parameters of the depth camera are obtained, including the focal length and principal point, as well as the extrinsic parameters, represented by the rotation matrix R_{CAM} and translation vector b. The intrinsic matrix is typically expressed as:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

where f_x and f_y are the focal lengths along the X-axis and Y-axis, respectively, and c_x and c_y represent the optical center of the image.

After obtaining the camera intrinsic parameters, each pixel in the depth map needs to be converted into 3D spatial coordinates. Given a pixel located at (u_d, v_d) in the depth map with a depth value of d, its corresponding 3D coordinates (X, Y, Z) in the camera coordinate system can be computed using the following equations:

$$X = \frac{(u_d - c_x) \cdot Z}{f_x} \tag{3}$$

$$Y = \frac{(v_d - c_y) \cdot Z}{f_y} \tag{4}$$

$$Z = d \tag{5}$$

Next, the obtained 3D point cloud is mapped onto the RGB image. The camera intrinsic parameters are used to transform the 3D coordinates back into pixel coordinates, while the camera extrinsic parameters are applied to adjust the 3D coordinates. The transformation is expressed as follows:

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = K \cdot (R_{CAM} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + b) \tag{6}$$

where u' and v' represent the mapped pixel coordinates.

After this step, the aligned point cloud data is mapped onto the segmentation mask generated by the semantic segmentation model. The RGB values of the point cloud are then modified based on the mask category corresponding to each point's location.

2) Map Construction

First, two coordinate systems are defined: the sensor coordinate system S and the map coordinate system M, as illustrated in Figure 3. The sensor coordinate system S is fixed to the depth camera, while the map coordinate system M is associated with S through a specified translation r_{SM} and rotation Φ_{SM}. Let $P_i = (x_i, y_i, \hat{h}_{si}, C_i)$ represent a grid cell i in the SEM, where x_i and y_i denote the position of grid cell i, $\hat{h}_{si}$ represents the estimated terrain height at grid cell i, and C_i indicates the semantic category of grid cell i.

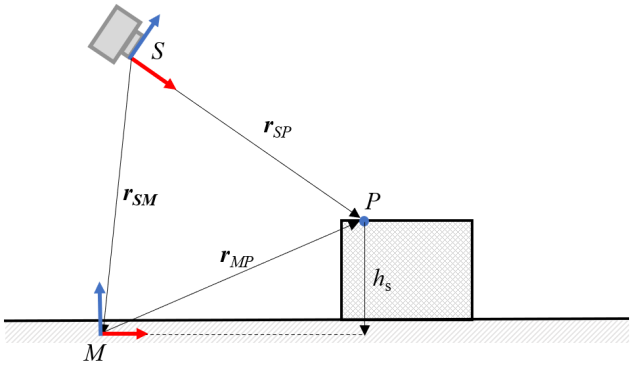


Figure 3. Coordinate system selationship.

In the map coordinate system M, the height measurement value $\tilde{p}$ is approximated by a Gaussian probability distribution as follows: $\tilde{p} \sim N(p, \sigma_p^2)$, where p represents the mean height estimate, and σ_p^2 denotes the variance. The depth camera continuously captures point cloud data of the terrain, and new measurements are processed as spatial points before being mapped onto the elevation map. A measurement point in the sensor coordinate system S is denoted as Sr_{sp}, and these points can be transformed into the corresponding height measurement value p using the following equation:

$$p = P \cdot (\Phi_{SM}^{-1}(Sr_{SP}) - Mr_{SM}) \tag{7}$$

where P = [0,0,1] is a projection matrix, mapping the 3D measurement values to the height in the map coordinate system. Φ_{SM} represents the rotation matrix from the sensor coordinate system S to the map coordinate system M. Mr_{SM} denotes the translation vector from the sensor coordinate system S to the map coordinate system M.

To obtain the variance σ_p^2 of the height measurement values, the Jacobian matrix of the sensor measurements J_S and the Jacobian matrix of sensor coordinate system rotation J_Φ are derived from the above equation.

$$J_S = \frac{\partial p}{\partial Sr_{SP}} = PC(\Phi_{SM})^T \tag{8}$$

$$J_\Phi = \frac{\partial p}{\partial \Phi_{SM}} = PC(\Phi_{SM})^T Sr_{SM}^{\times} \tag{9}$$

where $C(\Phi_{SM})$ represents the rotation matrix mapping from the sensor coordinate system S to the map coordinate system M, which describes the relative pose relationship between the two coordinate systems. The parameter $Sr_{SM}^{\times}$ denotes the vector r_{SM} from the sensor coordinate system S to the map frame M, and the cross-product symbol × is used to represent vector rotation.

The error propagation of the variance σ_p^2 is given by:

$$\sigma_p^2 = J_S \Sigma_S J_S^T + J_\Phi \Sigma_{\Phi S} J_\Phi^T \tag{10}$$

where Σ_S represents the covariance matrix derived from the sensor measurements, and $\Sigma_{\Phi S}$ denotes the covariance matrix

of sensor rotation. Both covariance matrices are obtained from the transformation matrix.

The new height measurement $(\tilde{p}, \sigma_p^2)$ is fused with the existing estimated value $(\hat{h}_s, \sigma_h^2)$ in the map using a one-dimensional Kalman filter, as follows:

$$\hat{h}_s^+ = \frac{\sigma_p^2 \hat{h}_s^- + \sigma_h^{2-} \tilde{p}}{\sigma_p^2 + \sigma_h^{2-}} \tag{11}$$

$$\sigma_h^{2+} = \frac{\sigma_h^{2-} \sigma_p^2}{\sigma_p^2 + \sigma_h^{2-}} \tag{12}$$

where $\hat{h}_s^-$ and σ_h^{2-} represent the height and variance before the update, while $\hat{h}_s^+$ and σ_h^{2+} denote the height and variance after the update. The initial value of σ_h^2 is primarily determined by the specific scenario, sensor resolution, and measurement accuracy. In this case, it is set to σ_h^2=0.01.

After obtaining the new height estimate for grid cell *i*, the RGB values *Color*(a_i) of the point cloud a_i within this cell are analyzed. The most frequently occurring color is recorded as the dominant color $Color_m$ of the grid cell.

$$Color_m = \arg\max(Color(a_i)) \tag{13}$$

The semantic category C_i of the grid cell is determined based on its dominant color $Color_m$. Ground and stairs are defined as traversable areas. For grid cells without a dominant color, i.e., when no single color accounts for more than 80% of the observed values, the cell is classified as an obstacle.

III. Experiments

A. Implementation

1) Experimental Environment: The experimental data in this study were collected using an Intel RealSense D455 depth camera. The processing platform consists of a laptop equipped with an Intel Core i9-14900HX CPU (32GB RAM) and an NVIDIA GeForce RTX 4060 GPU (8GB VRAM), running on Ubuntu 20.04. The semantic elevation mapping method was implemented based on the Grid Map library [10] and developed using a combination of C++ and Python.

2) Dataset: This study constructs a dataset comprising 800 images for indoor multi-person scenarios, such as subways, shopping malls, and hospitals. The dataset is divided into 500 images for training, 150 images for validation, and 150 images for testing. It includes 6,940 labeled objects across four categories: ground, stairs, pedestrians, and doors. Due to its relatively small scale, the image segmentation model was first pretrained on the COCO 2017 Train dataset [11] and then fine-tuned using the dataset from this study.

B. Performance Analysis

To validate the improved image segmentation model, experiments were conducted on three common targets in indoor cross-floor scenarios: ground, stairs, and pedestrians. The segmentation performance of YOLACT [9], YOLACT++ [12], and the proposed model was evaluated using the test set of the constructed dataset. The experimental results are summarized in Table I. For quantitative accuracy evaluation, Average Precision (*AP*) for single-class models and Mean Average Precision (*mAP*) for multi-class models are commonly used as segmentation performance metrics.

TABLE I. Comparison of Segmentation Results

Model	mAP	AP_{50}	AP		
			Ground	*Stair*	*Person*
YOLACT	42.85	67.15	84.82	77.48	63.85
YOLACT++	46.48	70.76	86.30	80.62	**68.73**
Ours	**46.51**	**71.20**	**90.43**	**81.77**	68.34

The results reveal that the enhanced image segmentation model outperforms the original models in terms of overall accuracy. However, due to similar textures and indistinct boundaries between the ground and stairs in indoor settings, segmentation errors are prone to occur when the depth camera captures these surfaces from a distance, resulting in suboptimal segmentation performance.

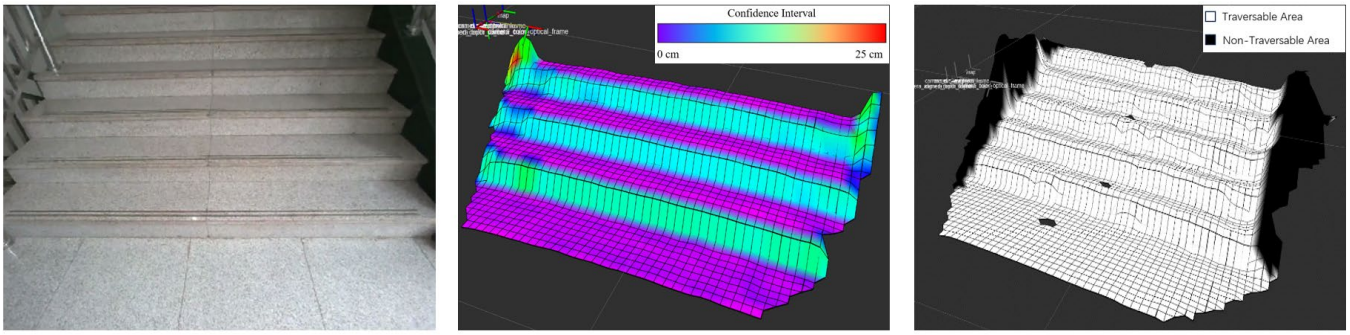
The segmentation results and point cloud fusion for a real-world stairway scenario are illustrated in Figure 4. In environments where certain obstacles are difficult to distinguish using geometric cues alone—such as the long wooden plank shown in Figure 4(a)—the segmentation model successfully identifies and classifies the object as an obstacle. These findings demonstrate that integrating semantic information into the elevation map effectively mitigates the limitations of purely geometry-based methods. Consequently, incorporating multimodal information into SEM is crucial for enhancing the practical applicability of robotic systems in real-world scenarios.

We evaluated the SEM's performance in an indoor stairway scenario where each step is 15 cm high, as shown in Figure 5. Figures 5(b) and 5(c) illustrate the elevation map and semantic map derived from the scene in Figure 5(a), respectively. Figure 6 presents the estimated confidence intervals obtained through map sampling. We observe that flat surfaces, such as the ground and stair treads, exhibit low confidence intervals, with a height estimation error of 3.1%. In contrast, stair edges show higher uncertainty due to significant height variations, resulting in larger confidence intervals in these areas. This comparison indicates that the SEM's height estimation accurately reflects the actual terrain, confirming the effectiveness of the proposed approach.

The computational efficiency of the SEM generation process was evaluated by averaging results over 300 algorithm iterations, as summarized in Table II. Each iteration required 69.36 ms, corresponding to an update rate of 13 Hz for the SEM. The semantic segmentation stage dominated the total processing time, given its pixel-level analysis and category filtering. Nevertheless, experimental results indicate that incorporating multimodal data into the elevation map maintains computational efficiency within the same order of magnitude, thereby meeting real-time requirements for practical applications.

(a) Point Cloud (b) Segmentation Mask (c) Semantic Point Cloud

Figure 4. Semantic point cloud generation results.



(a) Real-world scene (b) Height layer (c) Semantic layer

Figure 5. SEM construction results.

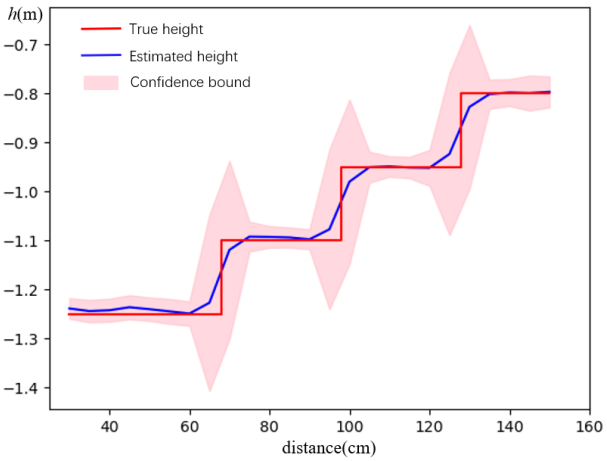


Figure 6. Stair height estimation results.

TABLE II. SEMANTIC ELEVATION MAP TIME PERFORMANCE

Process	Time(ms)
Semantic segmentation	26.59±2.33
Semantic point cloud generation	17.84±1.95
Height computation	13.72±1.48
Multimodal fusion	11.21±1.14
Total	69.36±5.71

IV. CONCLUSION

This paper introduces a real-time SEM construction method based on a hierarchical residual network. First, we enhance YOLACT by integrating hierarchical residual connections, thereby improving its ability to detect object boundaries in complex environments while maintaining a lightweight architecture. Next, we construct the SEM by aligning and fusing segmentation masks with point cloud data, effectively embedding semantic information into the elevation map to improve environmental representation. To evaluate the improved semantic segmentation model, we first test it on a self-constructed dataset, demonstrating its superior performance. We then assess the proposed approach in a real-world stairway scenario, where the results show a height estimation error of less than 3.1% and an update rate of 13 Hz. These findings confirm that the method can accurately represent both terrain height and its semantic categories, highlighting its suitability for a wide range of real-world applications.

REFERENCES

[1] Miki T, Wellhausen L, Grandia R, et al. Elevation mapping for locomotion and navigation using gpu[C]//2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2022: 2273-2280.

[2] Krüsi P, Furgale P, Bosse M, et al. Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments[J]. Journal of Field Robotics, 2017, 34(5): 940-984.

[3] Fankhauser P, Bloesch M, Hutter M. Probabilistic terrain mapping for mobile robots with uncertain localization[J]. IEEE Robotics and Automation Letters, 2018, 3(4): 3019-3026.

[4] Jenelten F, Grandia R, Farshidian F, et al. TAMOLS: Terrain-aware motion optimization for legged systems[J]. IEEE Transactions on Robotics, 2022, 38(6): 3395-3413.

[5] Mishra B, Calvert D, Bertrand S, et al. Efficient terrain map using planar regions for footstep planning on humanoid robots[C]//2024 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2024: 8044-8050.

[6] Maturana D, Chou P W, Uenoyama M, et al. Real-time semantic mapping for autonomous off-road navigation[C]//Field and Service Robotics: Results of the 11th International Conference. Springer International Publishing, 2018: 335-350.

[7] Ewen P, Li A, Chen Y, et al. These maps are made for walking: Real-time terrain property estimation for mobile robots[J]. IEEE Robotics and Automation Letters, 2022, 7(3): 7083-7090.

[8] Erni G, Frey J, Miki T, et al. MEM: Multi-Modal Elevation Mapping for Robotics and Learning[C]//2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2023: 11011-11018.

[9] Bolya D, Zhou C, Xiao F, et al. Yolact: Real-time instance segmentation[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 9157-9166.

[10] Fankhauser P, Hutter M. A universal grid map library: Implementation and use case for rough terrain navigation[J]. Robot Operating System (ROS) The Complete Reference (Volume 1), 2016: 99-120.

[11] Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. Springer International Publishing, 2014: 740-755.

[12] Bolya D , Zhou C , Xiao F ,et al. YOLACT++ Better Real-Time Instance Segmentation[J].IEEE transactions on pattern analysis and machine intelligence, 2022, 44(2): 1108-1121. DOI:10.1109/TPAMI.2020.3014297.

[13] Mu H, Zhang G, Ma Z, et al. Dynamic Obstacle Avoidance System Based on Rapid Instance Segmentation Network[J]. IEEE Transactions on Intelligent Transportation Systems, 2023.