

An effective method for UAV LiDAR point cloud data segmentation using SAL

Nguyen Thi Huu Phuong¹, Nguyen The Loc¹, Nguyen Thi Mai Dung¹, and Le Hong Anh¹

¹ Faculty of Information Technology, Hanoi University of Mining and Geology, Hanoi, Vietnam

nguyenthihuuphuong@humg.edu.vn

nguyentheloc@humg.edu.vn

nguyenthimaidung@humg.edu.vn

lehonganhanh@humg.edu.vn

Abstract. Segmentation of LiDAR point cloud data is a fundamental challenge in LiDAR data processing, directly impacting the accuracy of information retrieval. This paper introduces an enhanced SAL (Segment Anything in LiDAR) method for labeling LiDAR points without manual supervision, leveraging pseudo-labels to reduce the number of points required for model training. The proposed approach integrates point features and an encoder step using tokens generated from feature extraction and CLIP (Contrastive Language-Image Pre-training) embeddings. Experimental results on a real-world UAV LiDAR dataset collected in Hoa Binh, Vietnam, demonstrate segmentation accuracy exceeding 0.9, validating the method's effectiveness for semantic segmentation of UAV LiDAR point clouds.

Keywords: UAV LiDAR segmentation, SAL, pseudo-labeling, LiDAR point cloud, semantic segmentation.

1 Introduction

LiDAR (Light Detection and Ranging) provides accurate 3D spatial information, supporting applications from autonomous driving to environmental monitoring. When integrated with UAVs, LiDAR enables efficient data collection in complex and inaccessible terrains. However, raw LiDAR point clouds must be segmented into semantically meaningful components to support downstream tasks. Currently, with the development of UAVs (Unmanned Aerial Vehicles), integrating LiDAR technology onto UAV platforms has opened new avenues for space research and natural resource management. LiDAR data obtained from UAVs will also be in the form of point clouds. Through algorithmic processing, it is possible to separate canopy layers, determine forest heights, calculate tree volumes, and generate digital surface models (DSM), digital terrain models (DTM) and proves valuable in providing detailed service information vital for management-based decision-making.

In this paper, the authors focus on segmentation techniques for LiDAR point cloud. Point cloud segmentation is the process of dividing the points into different semantic

groups that correspond to objects on the surface, such as objects or structures within the environment. The goal of segmentation is to classify each point into a specific object category, such as “car,” “road,” “building,” or “tree,” based on its representation in the 3D scene [1], [2], [3]. This technique has seen significant improvements in accuracy and efficiency, thanks to advanced 3D sensors and deep learning algorithms, opening up applications in robotics, autonomous vehicles, and augmented reality [3]. The methods employed for LiDAR segmentation include point feature-based segmentation, interpolation decoding, machine learning, and point cloud subdivision. Recent deep learning-based models such as PointNet++, Cylinder3D, SPVCNN, and SqueezeSegV3 have demonstrated significant improvements in segmentation performance. Nevertheless, these models typically require retraining for new environments or object classes and are heavily dependent on manually annotated training data [1], [2]. To address these limitations, zero-shot and prompt-based segmentation models have emerged as promising alternatives. In this context, Segment Anything in LiDAR (SAL), a recently proposed framework by Ošep et al. [5], enables segmentation of raw LiDAR point clouds without the need for manual supervision or retraining. SAL combines pseudo-labeling with prompt-guided feature matching to perform flexible segmentation using point-, box-, or text-based prompts. However, its application to UAV LiDAR data has not been thoroughly explored in existing literature.

2 Related Work

LiDAR point cloud segmentation approaches can be grouped into three main categories: supervised learning, semi-supervised learning, and zero-shot or prompt-based segmentation.

Supervised deep learning models such as PointNet++, Cylinder3D, SPVCNN, and SqueezeSegV3 have shown excellent performance on benchmark datasets. However, these models require large volumes of labeled training data and are not scalable to new environments without retraining. Works such as Maria Axelsson et al. [2] and others [6 - 9] have applied these methods to UAV LiDAR with varying success.

Recently, prompt-based and zero-shot models have gained attention due to their ability to operate without manually labeled data. SAL [5] is one such framework that combines pseudo-labeling with flexible prompts. Other efforts such as Segment Anything 3D (SA3D) [16], GeoLLMs [15], and GeoFusion [14] show the potential of general-purpose geospatial models.

To the best of our knowledge, no prior work has applied the SAL framework specifically to UAV LiDAR data. Our work addresses this gap by adapting SAL to segment UAV LiDAR point clouds using point, box, and text prompts, evaluated on a real-world dataset from Hoa Binh, Vietnam. To address these limitations, our work explores the application of the SAL framework for semantic segmentation of UAV LiDAR point clouds. SAL offers a zero-shot capability, eliminating the need for extensive retraining and annotated datasets. To the best of our knowledge, this is the first study to adapt and evaluate the SAL framework for UAV LiDAR data. The pro-

posed approach combines pseudo-labeling techniques with point-based feature encoding, as described in Section 4.

3 Background

3.1 UAV LiDAR data and LiDAR point cloud segmentation overview

UAV LiDAR systems generate high-density three-dimensional (3D) point cloud data through laser scanning of real-world surfaces. These systems operate by emitting laser pulses that reflect off surfaces such as terrain, vegetation, and man-made structures, the returned signals are used to assign accurate geospatial coordinates to each point [10]. Compared to traditional surveying techniques, UAV-based LiDAR offers several advantages. It enables data collection over difficult-to-access or densely vegetated areas and provides detailed structural information of landscapes. The resulting point cloud data facilitates a wide range of applications including forestry, urban planning, digital elevation modeling (DEM/DSM), and natural resource monitoring.

LiDAR point cloud segmentation is a crucial step in transforming raw spatial data into structured information. It involves partitioning the point cloud into semantically meaningful regions corresponding to real-world objects such as buildings, roads, trees, or terrain surfaces. This process enables downstream tasks such as object detection, classification, mapping, and scene understanding [11]. LiDAR data is typically unstructured, irregularly spaced, and often contains noise or outliers. Variations in point density and occlusions from complex surfaces further complicate segmentation. The lack of explicit neighborhood information also hinders traditional spatial analysis methods. Current segmentation approaches are commonly categorized into three main types:

- Point-based methods: These techniques operate directly on raw point clouds using point-wise features such as elevation, intensity, ... While they avoid unnecessary transformations, their reliance on local neighborhood search can become computationally expensive, especially for large-scale datasets. PointNet and PointNet++ [12] operate directly on raw 3D coordinates, learning spatial features through shared multi-layer perceptrons and hierarchical sampling. These methods offer high accuracy on small datasets but tend to struggle with scalability in large scenes due to inefficient neighbor search.
- Voxel-based methods: In this approach, the point cloud is discretized into voxels (3D grids), allowing for regular data representation. Features are extracted using 3D convolutional neural networks (3D CNNs). Voxel-based methods typically achieve high accuracy but require significant memory and computational resources. Beside , research on SPVCNN [13] and Cylinder3D [17], discretize the point cloud into regular voxel grids, allowing the use of 3D convolutional neural networks (3D CNNs).
- Projection-based methods: These methods convert 3D point clouds into 2D images for processing using 2D convolutional networks. Although they offer improved computational efficiency, maintaining geometric consistency during projection and ensuring segmentation accuracy across diverse scenes remain challenging [14].

The development of Geospatial Foundation Models and GeoLLMs (Geospatial Large Language Models) has gained attention. Models such as GeoFusion [14], GeoLM [15], and Segment Anything 3D (SA3D) [16] integrate multimodal data—including satellite imagery, LiDAR, and textual metadata—for spatial reasoning and object-level understanding. GeoLLMs leverage large-scale pretraining on geospatial corpora to support zero-shot spatial querying, semantic segmentation, and map-based inference. Moreover, segmentation performance can vary significantly depending on the characteristics of the scanned area. For instance, ground points may be identified using elevation thresholds, while buildings and trees require the use of morphological or intensity-based features. Given these challenges, recent efforts have focused on incorporating learning-based techniques such as deep neural networks and attention mechanisms to improve segmentation performance in complex scenes. To mitigate these limitations, recent research has turned to zero-shot and prompt-based segmentation models. A notable example is the SAL framework [4], which performs semantic segmentation of raw point clouds by associating feature embeddings with various prompt types (point, box, text) without manual supervision or retraining. Our work contributes to this research direction by adapting the SAL framework, a zero-shot, prompt-based segmentation method, to UAV LiDAR point clouds. The use of the SAL framework for UAV LiDAR point cloud segmentation and proposes a customized segmentation pipeline that leverages both geometric and semantic prompt encoders. Building on the SAL model, we propose a UAV-based LiDAR point cloud segmentation approach that incorporates improvements in the proposed prompts. The input data is extracted based on various point features, such as coordinates, altitude, and intensity values. The proposed method was tested on a real UAV LiDAR dataset collected in Hoa Binh Province, Vietnam, achieving an accuracy of over 0.9. Our contributions are as follows:

1. We propose a SAL-based segmentation method tailored for UAV LiDAR point clouds, incorporating multiple prompt types (point, box, and text) and feature extraction using PointNet++ and CLIP embeddings.
2. We present a comprehensive evaluation of the proposed method on a real-world UAV LiDAR dataset collected in Hoa Binh province, Vietnam, with segmentation accuracy exceeding 0.9 across multiple object classes.
3. We compare the segmentation performance of our SAL-based method with state-of-the-art supervised approaches such as PointNet++ and Cylinder3D, demonstrating its competitiveness despite requiring no labeled data or model retraining.
4. We position SAL within the broader context of emerging prompt-based and geospatial foundation models (e.g., GeoLLMs), discussing its lightweight nature and suitability for deployment in resource-constrained environments.

3.2 LiDAR segmentation with SAL

SAL is a recently proposed framework by Aljoša Ošep et al., introduced in 2024 under NVIDIA Research and published on arXiv [4]. The core idea of SAL is to enable semantic segmentation of raw LiDAR point clouds without transforming them into other representations such as 2D images or voxel grids. Instead, SAL directly

processes raw 3D data through prompt-based segmentation using pseudo-labels and zero-shot models. Zero-shot learning, also referred to as open-vocabulary recognition, enables the model to assign labels to unseen classes during inference. This is achieved by leveraging visual-linguistic embedding spaces, such as those derived from CLIP [18, 19]. In this setup, only a subset of object classes is seen during training, while the model learns to generalize to unseen classes via shared semantic features. In the SAL framework, LiDAR panoptic segmentation (LPS) is performed on the entire raw point cloud using smooth pseudo-labels. This allows segmentation to be applied even in noisy data environments [20]. The process includes attaching soft pseudo-labels to each point, enabling the system to predict object labels across the entire point cloud efficiently.

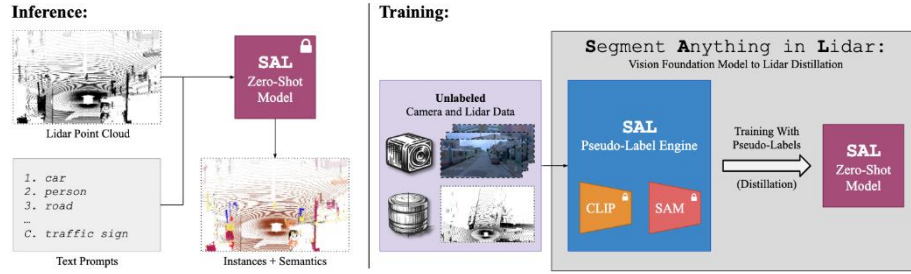


Fig. 1. LiDAR point cloud segmentation with SAL (Source: Research Nvidia)

As illustrated in Fig. 1, the segmentation pipeline of SAL includes three core components:

1. SAM (Segment Anything Model): Used to estimate pseudo-smooth masks for objects in the point cloud, based on spatial coherence.
2. CLIP-based Label Estimation: Semantic features of each pseudo-labeled region are computed using CLIP embeddings, allowing the model to infer object categories from text or prompt descriptors.
3. Calibration Sensor Alignment: A spatial alignment process is applied to transfer the labels and embeddings from the CLIP space into the LiDAR point cloud domain.

Using the extracted pseudo-labels, the SAL model constructs a training dataset suitable for zero-shot learning. It leverages a discrete convolutional encoder and a decoding module to compute weighted segmentation scores and mask predictions. The decoder matches extracted features with the prompt embeddings to identify and label object instances within the point cloud. An important characteristic of the SAL framework is that it does not require model retraining when deployed on new datasets or new semantic classes. Thanks to its zero-shot capability, SAL can generalize to different segmentation tasks and input modalities without fine-tuning. This property offers significant advantages in practical scenarios, especially when working with large-scale or dynamic 3D environments. In summary, the SAL approach represents a shift from traditional supervised learning-based segmentation to a more flexible and generalizable zero-shot paradigm. Its ability to directly segment raw point clouds using soft pseudo-labels and diverse prompts makes it highly suitable for applications

in autonomous vehicles, robotics, geospatial analytics, and smart environment systems.

4 Proposed methodology

The proposed segmentation process is shown in Figure 2 below.

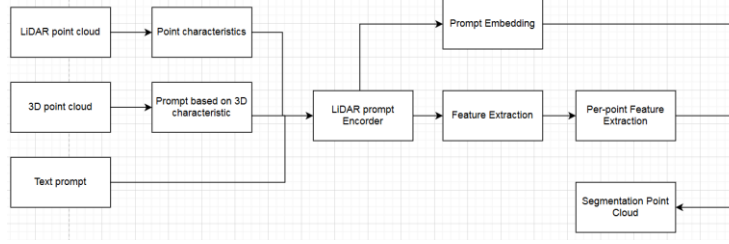


Fig. 2. The proposed segmentation procedure

The segmentation process consists of four main stages:

1. Input and Prompt Initialization

The input consists of raw UAV LiDAR point clouds collected from the survey site, along with point-level attributes such as:

- + Characteristics of point: Spatial coordinates (x, y, z); Intensity values; Pulse return number; Reflectance characteristics.

- + Prompt: Three types of prompts are optionally utilized:

- Point prompts: Sampled key points that represent unique regions in the data (e.g., high elevation, strong reflectance).

- 3D prompts: Geometric prompts representing volumes containing target objects.

- Text prompts (optional): Natural language descriptors (e.g., "power line", "high tree", "terrain point") that provide semantic cues.

2. Prompt Encoding

Each prompt type is encoded independently to create embeddings that serve as semantic guidance in the segmentation process:

- Point Prompt Encoder: Converts spatial coordinates into feature vectors via a shallow neural network.

- Box Prompt Encoder: Extracts geometric features (e.g., center, dimensions) and embeds them as region descriptors.

- Text Prompt Encoder: Uses a transformer or pre-trained CLIP model to convert natural language labels into dense embeddings.

All prompt embeddings are aggregated into a unified prompt token set $\mathcal{P} = \{p_1, p_2, \dots, p_m\}$ which is used in the decoding stage.

3. Point Cloud Feature Extraction

Features from the point cloud are extracted using either PointNet++ or a transformer-based encoder such as SFormer:

- PointNet++: A hierarchical network that learns local and global features directly from raw 3D coordinates.

- SFormer: A sparse attention-based architecture optimized for LiDAR data with non-uniform point distribution.

Vector F is the feature of N points in the point cloud, $F = \{f_1, f_2, \dots, f_N\}$ with $f_i \in \mathbb{R}^d$. Set of prompt embeddings $= \{p_1, p_2, \dots, p_m\}$ with $p_j \in \mathbb{R}^d$. The decoder's task is to combine F and P to find a soft mask with value 0 or 1 for each point p_i . After having the combined feature, we use MLP to return the probability:

$$\hat{y}_i = \sigma \left(MLP(f'_i) \right) \in [0, 1] \quad (1)$$

And if $\hat{y}_i \geq 85\%$, the point will be labeled as belonging to the specified object class.

4. Prompt-aware Mask Decoding

Each stage is detailed below.

- Input:

- LiDAR Prompt Encoder

Each prompt type is encoded separately according to the prompt characteristics and the collected input data set of the object.

Point Prompt $\rightarrow$ Converts the prompt coordinates into feature vectors (embeddings).

Box Prompt $\rightarrow$ Encode by getting geometric features such as center position, size.

Text Prompt $\rightarrow$ Use CLIP or transformer encoders to get description vectors from text.

All these prompt embeddings are combined into a Prompt Token for use in the decoder.

- Point Cloud Feature Extractor

The goal of this step is to extract features for the point cloud to proceed to extract features for each point in the point cloud. In this step, use PointNet++ or SFormer to extract features. With:

PointNet++: Train the model directly from the coordinates of the points.

SFormer: Combining attention and sparse convolution – more powerful with sparse LiDAR data.

The result of the feature extraction step is a set of feature vectors, each corresponding to a point.

- Prompt-aware Mask Decoder

This is the step to label points during segmentation. We combine the prompt token and the point feature vector through attention (similar to the mechanism in the transformer). At this time, the model selects the point that matches the input prompt to label the point. Each output point will have a probability of belonging to the described object.

At that time, a mask determines which point in the point cloud belongs to the described object.

Suppose, we have:

Vector F is the feature of N points in the point cloud, $F = \{f_1, f_2, \dots, f_N\}$ with $f_i \in \mathbb{R}^d$. Set of prompt embeddings $P = \{p_1, p_2, \dots, p_M\}$ with $p_j \in \mathbb{R}^d$. The decoder's task is to combine F and P to find a soft mask with value 0 or 1 for each point p_i .

If $\hat{y}_i \geq 85\%$, the point will be labeled on the class object.

- Segmentation Output:

The final segmentation output consists of soft-labeled (value 0 -1) point clouds, where each point is tagged with its predicted semantic class. These results can be visualized directly in 3D or used as input for further analysis such as object modeling, classification, or map generation.

5 Experiment and result

5.1 Experiment dataset

The proposed segmentation method was evaluated using a UAV LiDAR dataset collected in Hoa Binh province, a mountainous region in northern Vietnam. The data acquisition on August 1–2, 2024, focusing on separating power lines from surrounding vegetation and terrain features. The dataset consists of 6,012,700 3D points, with an average density of 23.2 points per square meter and a maximum of five return pulses per emitted laser. Each point includes attributes such as spatial coordinates (X, Y, Z), intensity, and elevation. Figures 3 to 5 illustrate the spatial distribution, elevation characteristics, and 3D structure of the collected point cloud.

Extent X:	0.00 : 298334.00
Extent Y:	-97303.00 : 15510.00
Extent Z:	-683.00 : 36222.00

Fig. 3. X, Y, Z of point

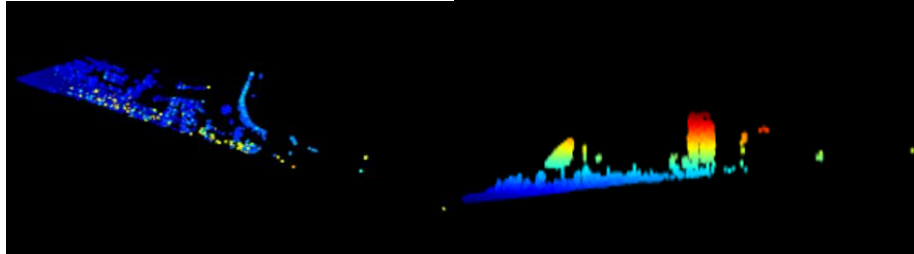


Fig. 4. Intensity and elevation color of point cloud

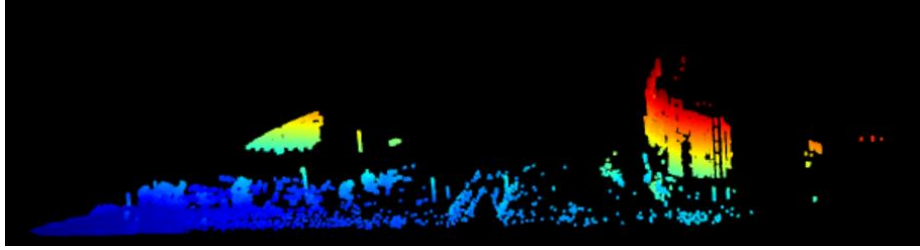


Fig. 5. 3D point cloud

5.2 Experiment result

Pseudo-labels were generated using the prompt-based encoder described in Section 5. The selected text prompts included: high tree, power line, terrain point, low vegetation, pole.

Each prompt was encoded into semantic embeddings using CLIP and combined with point-wise features extracted from PointNet++. These embeddings were used to assign soft pseudo-labels to individual points. These were integrated through the SAL prompt-aware mask decoder to assign soft pseudo-labels to each point. The labeling performance on a selected subregion is summarized in Table 1

Table 1 Accuracy of pseudo-label

	High tree	Powerline	Terrain point	Low tree	Pole
Accuracy	0.89	0.93	0.89	0.9	0.89

The assigned pseudo-labels were then used as training data for the SAL framework. By learning the relationship between prompts and point cloud features, SAL effectively segmented the entire dataset based on the provided semantic descriptions. The assigned pseudo-labels served as weak supervision for the SAL segmentation model, which was trained without human-annotated ground truth.

The segmentation model was evaluated based on its precision, F1 score, and convergence rate during both training and inference. The results are presented in Table 2.

Table 2 Accuracy of training and labeling steps for points

	Precision	F1	Converge
SAL training	0.94	0.93	0.000023
Labeling	0.9	0.90	0.001

The results demonstrate that the proposed method achieves high segmentation accuracy without requiring manually annotated labels. The SAL framework effectively learns to map semantic prompts to meaningful 3D structures, and generalizes well across different object classes. The high F1 scores confirm that the model maintains a balanced trade-off between precision and recall, ensuring reliable object classification in complex scenes. Visual inspection of the segmented 3D model reveals that the

method successfully distinguishes between trees, terrain, poles, and power lines. The segmentation masks align closely with the underlying geometry, even in areas with high point density and occlusions. This demonstrates the practical effectiveness of the proposed method in real-world environments, particularly for applications such as vegetation monitoring and power infrastructure inspection. Visual inspection of the segmented 3D model reveals that the method successfully distinguishes between trees, terrain, poles, and power lines. The segmentation masks align closely with the underlying geometry, even in areas with high point density and occlusions. This demonstrates the practical effectiveness of the proposed method in real-world environments, particularly for applications such as vegetation monitoring and power infrastructure inspection.

To assess the effectiveness of the proposed method, we compared the SAL-based approach with two widely adopted supervised segmentation models:

PointNet++ [12]: a point-based model that processes raw point coordinates with local feature aggregation.

Cylinder3D [17]: a voxel-based model using cylindrical partitioning and 3D sparse convolution.

Both models were trained using a manually annotated subset of the Hoa Binh dataset, while SAL was applied using only pseudo-labels and zero-shot prompts.

Table 3 Method Precision, Accuracy, F1 Training data Pretraining of SAL with PointNet++ and Cylinder3D

Method	Precision	Accuracy	F1	Training data	Pretraining
PointNet++	0.9	0.91	0.9	Yes	Yes
Cylinder3D	0.91	0.9	0.9	Yes	Yes
SAL proposed	0.94	0.92	0.93	No	No

To benchmark our method, we compared the segmentation results against two state-of-the-art models: PointNet++ and Cylinder3D, both of which were trained on a manually annotated subset of the same dataset. Our SAL-based approach, in contrast, required no labeled training data. As shown in Table 3, SAL outperformed these models in accuracy and F1-score, particularly in distinguishing between “high tree” and “low vegetation” classes. The results indicate that our SAL method achieves superior segmentation accuracy compared to state-of-the-art supervised models, despite not requiring labeled training data or retraining. This demonstrates the strength of prompt-based zero-shot learning in practical, resource-constrained UAV LiDAR scenarios. SAL accurately differentiates between “high tree” and “low vegetation” classes based on elevation and canopy density, even in the presence of overlapping branches and occlusions. In contrast, PointNet++ often exhibits over-segmentation, and Cylinder3D tends to smooth out height variations due to voxelization effects. SAL, by leveraging point-level attention and prompt embeddings, maintains the spatial integrity of these features. Visual inspection shows that SAL consistently traces the full extent of power lines across varied terrain, whereas Cylinder3D partially misses or merges them with nearby vegetation. Additionally, SAL maintains consistent segmentation boundaries. This is due to its zero-shot prompt alignment strategy, which decouples segmentation performance from scene-specific training data. In

contrast, PointNet++ exhibits performance degradation in these areas, likely due to insufficient generalization from the training subset.

6 Conclusion

This paper presents an effective segmentation method for UAV LiDAR point clouds based on the Segment Anything in LiDAR (SAL) framework. By leveraging pseudo-labeling and zero-shot learning, the proposed approach eliminates the need for manual annotation and model retraining, making it highly adaptable to new datasets and real-world environments.

The segmentation pipeline integrates multiple types of prompts point-based, geometric (3D model), and text-based descriptors—together with feature extraction techniques such as PointNet++ and CLIP embeddings. Experimental results on real UAV LiDAR data collected in Hoa Binh, Vietnam, demonstrate the robustness of the method, achieving over 90% accuracy in semantic segmentation tasks across various object classes, including terrain, vegetation, power lines, and poles. The proposed method showcases several advantages: (1) it supports flexible, prompt-driven segmentation; (2) it does not rely on large-scale labeled datasets; and (3) it is scalable to diverse applications such as environmental monitoring, infrastructure inspection, and autonomous mapping. Future work will focus on enhancing the efficiency of the segmentation pipeline for real-time applications and evaluating the model's generalization performance on other UAV LiDAR datasets with varying landscape complexities. Additionally, integrating the SAL-based segmentation framework into operational systems for smart city and environmental management is a promising direction.

References

1. synopsis, "What is LiDAR?," 2024. [Online]. Available: <https://www.synopsys.com/glossary/what-is-lidar.html>.
2. Maria Axelsson*, Max Holmberg, Sabina Serra†, Hannes Ovren, Michael Tulldahl, "Semantic labeling of lidar point clouds for UAV applications", CVPR 2021
3. FARO, "Point Clouds generated from Lidar for Beginners," 2024. [Online]. Available: <https://www.faro.com/en/Resource-Library/Article/Point-Clouds-for-Beginners>.
4. BasicAI, "Introduction to 3D Point Cloud Segmentation," 26 4 2024. [Online]. Available: <https://medium.com/@BasicAI-Inc/3d-point-cloud-segmentation-guide-a073b4a6b5f3>.
5. Aljoša Ošep et al, "Better Call SAL: Towards Learning to Segment Anything in Lidar," arXiv, 2024.
6. Dubrovin, I., Fortin, C., & Kedrov, A. (2024), "An open dataset for individual tree detection in UAV LiDAR point clouds and RGB orthophotos in dense mixed forests". Scientific Reports, 14, Article 21938.

7. RIEGL Unveils Advanced UAV LiDAR Sensor VUX-100-25 and RiLOC-F at INTERGEO. Unmanned Systems Technology
8. Xinye Li, Ding Chen, "A survey on deep learning-based panoptic segmentation," *Digital Signal Processing*, 2022.
9. Hamid Daghighi et al, "A critical review of discontinuity plane extraction from 3D point cloud data of rock mass surfaces," *Computers & Geosciences*,, 2022.
10. Yanwei Fu, Tao Xiang, Yu-Gang Jiang, Xiangyang Xue, Leonid Sigal, Shaogang Gong, "Recent Advances in Zero-shot Recognition," *Computer Vision and Pattern Recognition*, 2017.
11. A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin,, "Learning transferable visual models from natural language supervision," *Int. Conf. Mach.*, pp. 1-33, 2021.
12. C. Qi et al., "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," in *NeurIPS*, 2017.
13. T. Tang et al., "SPVNAS: Sparse Voxel and Point Network for Semantic Scene Completion," in *ECCV*, 2020.
14. J. Cao et al., "GeoFusion: Foundation Models for Real-World Geospatial Understanding," *arXiv preprint arXiv:2310.06237*, 2023.
15. Z. Zhang et al., "GeoLM: A Foundation Model for Geospatial Artificial Intelligence," in *ACM SIGSPATIAL*, 2023.
16. H. Chen et al., "Segment Anything 3D: Zero-Shot 3D Segmentation with Multi-Modal Prompts," *arXiv preprint arXiv:2306.00988*, 2023.
17. Z. Zhuang et al., "Cylinder3D: An Effective 3D Framework for Driving Scene Understanding," in *CVPR*, 2021.
18. Jinyuan Shao, Ayman Habib, Songlin Fei, "SEMANTIC SEGMENTATION OF UAV LIDAR DATA FOR TREE PLANTATIONS", *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Volume XLVIII-1/W2-2023 ISPRS Geospatial Week 2023, 2–7 September 2023, Cairo, Egypt
19. Cailian Zhang, Chenweng Song et al, Individual tree segmentation from UAS Lidar data based on hierarchical filtering and clustering, *International Journal of Digital Earth*, Volume 17, 2024 - Issue 1
20. Dening Lu, Jun Zhou, Kyle Gao, Linlin Xu, Jonathan Li, 3D Learnable Super-token Transformer for LiDAR Point Cloud Scene Segmentation, *Computer Vision and Pattern Recognition*