agriculture

MDPI

Article

Simultaneous Localization and Mapping System for Agricultural Yield Estimation Based on Improved VINS-RGBD: A Case Study of a Strawberry Field

Quanbo Yuan [1,2], Penggang Wang [2], Wei Luo [2,3,4,*], Yongxu Zhou [2,3,4], Hongce Chen [2] and Zhaopeng Meng [1]

[1] College of Intelligence and Computing, Tianjin University, Tianjin 300350, China; yqb_cs2018@tju.edu.cn (Q.Y.); mengzp@tju.edu.cn (Z.M.)
[2] North China Institute of Aerospace Engineering, Langfang 065000, China; wangpg@stumail.nciae.edu.cn (P.W.); zhouyx@nciae.edu.cn (Y.Z.); chc_qqmail@stumail.nciae.edu.cn (H.C.)
[3] Aerospace Remote Sensing Information Processing and Application Collaborative Innovation Center of Hebei Province, Langfang 065000, China
[4] National Joint Engineering Research Center of Space Remote Sensing Information Application Technology, Langfang 065000, China
* Correspondence: luowei@radi.ac.cn

Abstract: Crop yield estimation plays a crucial role in agricultural production planning and risk management. Utilizing simultaneous localization and mapping (SLAM) technology for the three-dimensional reconstruction of crops allows for an intuitive understanding of their growth status and facilitates yield estimation. Therefore, this paper proposes a VINS-RGBD system incorporating a semantic segmentation module to enrich the information representation of a 3D reconstruction map. Additionally, image matching using L_SuperPoint feature points is employed to achieve higher localization accuracy and obtain better map quality. Moreover, Voxblox is proposed for storing and representing the maps, which facilitates the storage of large-scale maps. Furthermore, yield estimation is conducted using conditional filtering and RANSAC spherical fitting. The results show that the proposed system achieves an average relative error of 10.87% in yield estimation. The semantic segmentation accuracy of the system reaches 73.2% mIoU, and it can save an average of 96.91% memory for point cloud map storage. Localization accuracy tests on public datasets demonstrate that, compared to Shi–Tomasi corner points, using L_SuperPoint feature points reduces the average *ATE* by 1.933 and the average *RPE* by 0.042. Through field experiments and evaluations in a strawberry field, the proposed system demonstrates reliability in yield estimation, providing guidance and support for agricultural production planning and risk management.

Keywords: crop yield estimation; semantic segmentation; VINS-RGBD; Voxblox

Citation: Yuan, Q.; Wang, P.; Luo, W.; Zhou, Y.; Chen, H.; Meng, Z. Simultaneous Localization and Mapping System for Agricultural Yield Estimation Based on Improved VINS-RGBD: A Case Study of a Strawberry Field. *Agriculture* **2024**, *14*, 784. https://doi.org/10.3390/agriculture14050784

Academic Editors: Changyuan Zhai, Ning Wang and Jianfeng Zhou

Received: 28 March 2024
Revised: 11 May 2024
Accepted: 15 May 2024
Published: 19 May 2024

1. Introduction

In the face of challenges such as escalating global population, rural impoverishment, and the management of natural resources [1], the imperative for farmers to implement more sustainable practices to bolster both crop productivity and provision has never been more critical [2]. The realm of smart agriculture, bolstered by advancements in artificial intelligence, is witnessing a gradual maturation of technologies aimed at enhancing agricultural yields. Within this context, orchards represent a crucial segment of smart agricultural practices, experiencing continuous evolution [3]. To achieve intelligent management of orchards and accomplish tasks such as monitoring tree growth [4,5], yield estimation [6,7], and assisting agricultural robots in harvesting or spraying [8,9], constructing a real-time three-dimensional semantic reconstruction system is indispensable.

Due to its capabilities for extensive localization and mapping, visual simultaneous localization and mapping (SLAM) technology has seen significant application in orchard management, particularly with UAVs and unmanned ground vehicles (UGVs). Li et al. [10]

proposed a tree trunk extraction method based on the improved iForest algorithm and combined it with SLAM technology to achieve more accurate 3D reconstruction of roadside trees. Sukvichai et al. [11] introduced an ORB SLAM system that generated maps of greenhouse tomato fields by integrating unmanned aerial vehicles. Ramirez et al. [12] presented a dense point cloud reconstruction system by combining the RTAB-Map algorithm with drones, enabling the creation of 3D maps of vegetation indices for vegetation segmentation. Gimenez et al. [13] proposed an RGB-D data-based SLAM system for tree trunk detection which performs well even under low-light conditions. Mitrofanova et al. [14] integrated rtabmap SLAM with agricultural robots, allowing the robots to adapt better to changing agricultural environments. Meyer et al. [15] proposed a method for fusing camera and lidar data, achieving watermelon classification through 3D reconstruction. The aforementioned studies primarily focus on optimizing data processing algorithms and improving visual SLAM or 3D reconstruction capabilities. However, the constructed 3D maps are mainly limited to geometric information, lacking semantic information. Therefore, it is necessary to integrate semantic information into the maps to enrich map information and expand the application scope of maps.

Currently, semantic mapping is also being applied in the field of agriculture. Yuan et al. [16] developed a robust semantic SLAM system based on semantic features such as corn stalks and ground, used for monitoring the growth status of corn fields throughout the season. Pan et al. [17] proposed a novel semantic mapping and navigation framework using a 3D detection network architecture, enabling the autonomous navigation of agricultural robots. Wei et al. [18] combined visual SLAM technology with the BiSeNetV2 semantic segmentation network to propose a robust semantic SLAM system in dynamic scenes. Dong et al. [19] achieved improved pose estimation and mapping quality by extracting semantic features of corn stalks and applying them to robot navigation. Liu et al. [20] combined a semantic segmentation module with visual SLAM to achieve judgment of strawberry ripeness and recognition of fruit positions. The reconstruction of semantic maps mentioned above is primarily non-real time. However, real-time localization and semantic mapping are extremely important in agricultural scenarios. Whether it is harvesting robots or spraying robots, real-time perception of the surrounding environment is necessary for accurate task completion.

Real-time SLAM has also been preliminarily applied in smart agriculture. Yan et al. [21] proposed a real-time SLAM system based on multi-sensor fusion, achieving precise pose estimation and dense mapping in complex greenhouses. Islam et al. [22] introduced a real-time visual SLAM system based on RGB-D data that was capable of operating in complex agricultural environments with high accuracy. Li et al. [23] presented a real-time PL-F-SLAM system based on the ORB-SLAM2 framework and point-line features, enabling accurate mapping in low-texture agricultural environments. Zhang et al. [24] proposed a multi-sensor fusion SLAM framework which can be mounted on unmanned vehicles for real-time positioning and mapping in orchards. Liu et al. [25] introduced an ORB-Livox system utilizing YOLOv5 for detection, achieving real-time fruit localization. The above research is still lacking in positioning accuracy, and more accurate positioning accuracy plays a crucial role in establishing more accurate maps. The selection of high-quality feature points and the combination of multi-sensor data fusion can effectively improve positioning accuracy.

In this study, we propose a model based on an improved VINS-RGBD system for 3D semantic map reconstruction of strawberry orchards. The original Shi–Tomasi corner points in VINS-RGBD are replaced with L_SuperPoint feature points. A lightweight semantic segmentation network, PP-LiteSeg-T, is utilized for the semantic segmentation of the images; a radius outlier removal filter for outlier elimination from the point cloud; and Voxblox for storing and representing the point cloud map. To evaluate our proposed system, experiments designed to be conducted in real strawberry orchards are detailed in Section 3, aiming to verify the feasibility of the overall system. The main contributions of this study include the following:

1. We propose a 3D semantic reconstruction system that integrates an improved VINS-RGBD system with the PP-LiteSeg-T semantic segmentation network. This system fuses point cloud information into the map, addressing the issue of incomplete information representation in traditional point cloud maps and enhancing map applicability.
2. In the VINS-RGBD system's feature extraction module, the end-to-end lightweight L_SuperPoint feature points are employed to extract features from strawberry images. This significantly mitigates the problem of feature point tracking loss in complex strawberry orchard environments and simultaneously enhances localization accuracy.
3. We design a point cloud data postprocessing method that combines a radius outlier removal filter (hereinafter referred to as the radius filter) with Voxblox. This approach not only improves the accuracy of point cloud data and map construction but also significantly reduces the memory requirements for point cloud storage, effectively addressing the challenge of storing three-dimensional maps in large scenes.

The structure of this document is outlined as follows: The approach that we developed is detailed in Section 2. Section 3 presents the experiments and results designed to evaluate the effectiveness of our method, while Section 4 provides discussions on the findings. This paper concludes with a summary of the findings in Section 5.

2. Materials and Methods

2.1. Data Collection and Processing

Langfang City, located in the east–central part of Hebei Province, China (38°28′–40°15′ N, 116°7′–117°14′ E), is situated in the middle of the country. The strawberry picking season here extends from March to May each year. On 16 March 2024, photographs of strawberries were captured in a strawberry garden using an Intel RealSense D435i video camera and a Sony CX450 video camera. The D435i was utilized for taking wide-angle photographs of the strawberry plants, whereas the Sony CX450 was employed for close-up shots of the strawberry fruits. The experimental setup is depicted in Figure 1. The D435i camera was connected to a DELL computer via USB, and the images were recorded with an Ubuntu system using the Rosbag function of ROS; the experimental scene is illustrated in Figure 2.

Figure 1. Experimental equipment.

When preparing the training dataset, we used the Rosbag function to select 752 distant view images with a resolution of 640 × 480. Additionally, we captured 350 close-up images with a resolution of 1920 × 1080 using a Sony CX450 camera. When recording data for strawberry plants using the Rosbag function, the recorded data are shown in Table 1. The "/camera/aligned_depth_to_color/image_raw" topic represents the depth image, "/camera/color/image_raw" represents the RGB image, and "/camera/imu" represents the IMU data. The final dataset comprised 1102 images, all adjusted to a resolution of 640 × 480 for training. In the context of semantic segmentation, to simplify semantic annotation, objects within the images were classified into three categories: fruits, leaves, and background, the latter representing parts of the scene not occupied by strawberry plants. The annotation process was conducted using the LabelMe 3.16.7 software [26], which is a web-based image annotation tool developed by the Computer Science and

Artificial Intelligence Laboratory at the Massachusetts Institute of Technology (MIT). It enables online image annotation. Figure 3 shows an annotation example.

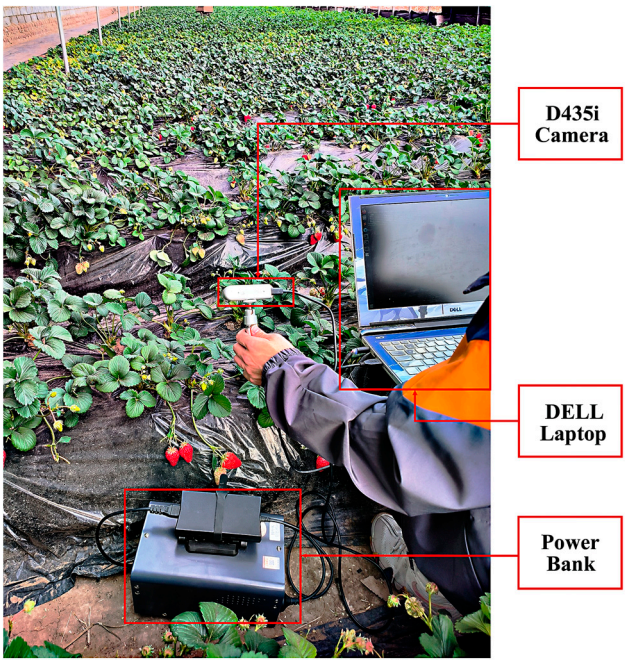


Figure 2. Schematic diagram of data collection scenario.

Table 1. Dataset information. "/camera/aligned_depth_to_color/image_raw" represents the topic for depth images published by the D435i camera, "/camera/color/image_raw" represents the topic for RGB images, and "/camera/imu" represents the IMU topic. The images are captured at a resolution of 640 × 480.

Dataset	/camera/aligned_depth_to_color/image_raw (msgs)	/camera/color/image_raw (msgs)	/camera/imu (msgs)
Strawberry Dataset 1	3030	3034	14,986
Strawberry Dataset 2	2300	2311	15,976
Strawberry Dataset 3	4006	4013	22,860
Strawberry Dataset 4	3160	3169	16,060
Strawberry Dataset 5	2965	2972	15,448

(**a**) (**b**)

Figure 3. The strawberry semantic segmentation dataset was constructed using Labelme for data annotation. In the image annotation, three categories were delineated: fruit (in red), foliage (in green), and background (in black). (**a**) represents the original strawberry image, while (**b**) represents the annotated strawberry image.

2.2. Method

Figure 4 illustrates the comprehensive process for conducting real-time localization and mapping within strawberry orchards. Initially, the system collects RGB and depth

images along with IMU data as inputs. Following this, the enhanced VINS-RGBD system undertakes the initial step by processing these images through a visual front-end, establishing the camera's initial orientation, and forwarding keyframes to the PP-LiteSeg-T network for detailed pixel-level semantic segmentation. The PP-LiteSeg-T network then processes these images to extract semantic information, which is subsequently integrated back into the VINS-RGBD system alongside the point cloud data. This integration is followed by a back-end optimization phase to refine the camera's positioning and enhance the semantic point cloud's accuracy. Lastly, this enriched semantic point cloud undergoes further processing through Voxblox ROS for the generation of Voxblox maps and a radius filter to eliminate any outliers, thus finalizing the semantic point cloud map creation.

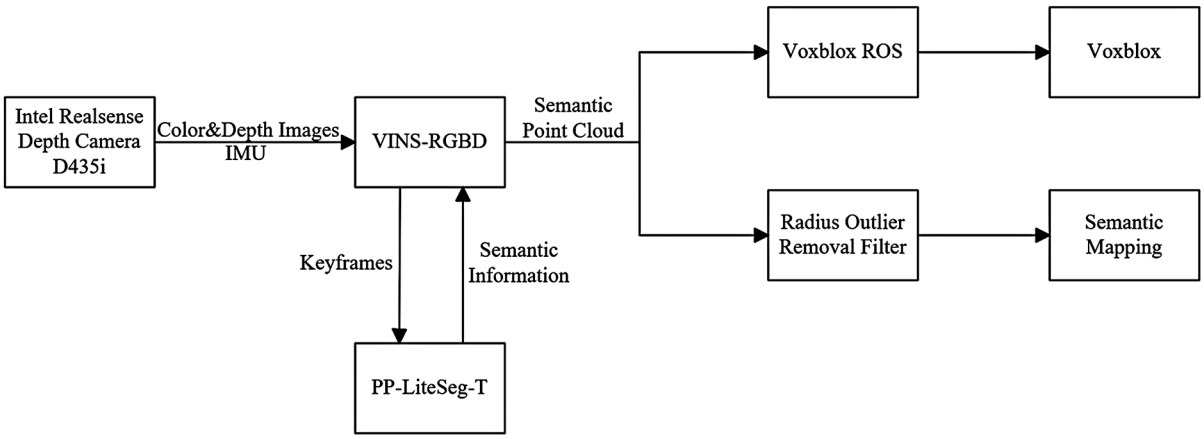


Figure 4. Overall methodology flowchart.

2.2.1. Semantic Segmentation Algorithm

In the conducted research, the PP-LiteSeg-T network emerges as an innovative and streamlined semantic segmentation framework tailored for the detailed analysis of strawberry orchard imagery. This advanced model adopts an encoder–decoder configuration that is significantly enhanced by the integration of three key components: a Flexible Lightweight Decoder (FLD), a Unified Attention Fusion Module (UAFM), and a Simple Pyramid Pooling Module (SPPM), as illustrated in Figure 5. We have included the visual explanations of the three key modules in Appendix A for reference.

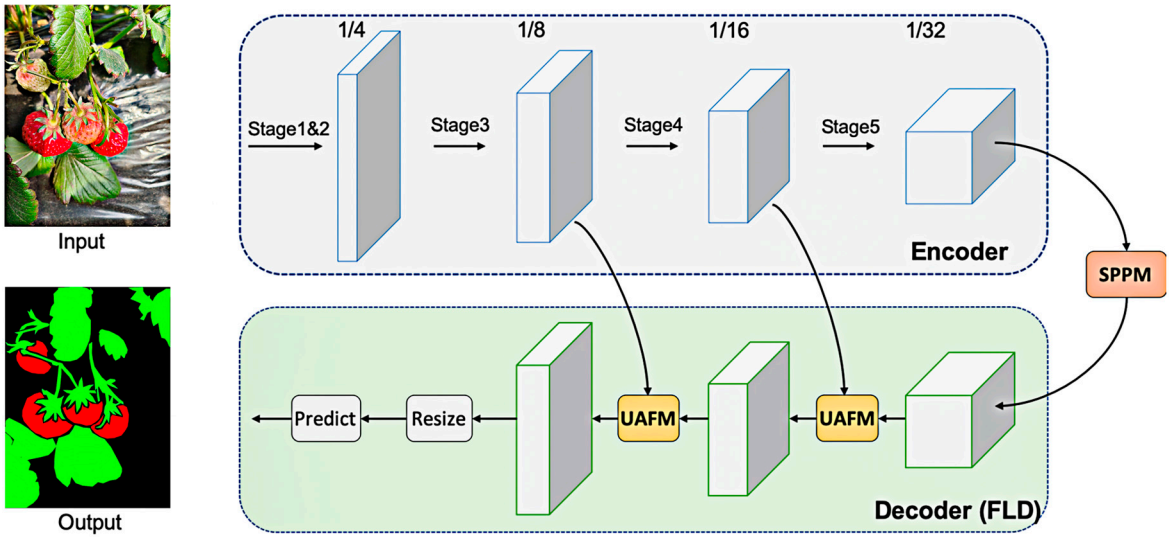


Figure 5. PP-LiteSeg-T network overall architecture.

The decoder, distinguished by its flexibility and lightweight nature, adeptly extracts features across a spectrum from basic to sophisticated levels. This progression is marked by a gradual expansion in the dimensionality of feature maps coupled with a strategic reduction in the number of channels. Such an approach ensures a meticulous balance in the computational load across the model's layers, effectively curtailing redundant processing efforts and thereby elevating the efficiency of the decoding process beyond that of traditional decoder designs.

The network's selection of a spatial attention mechanism within the UAFM serves to significantly refine the representation of features. This is achieved by upsampling the high-level feature map, denoted as F_{high}, to produce an upscaled version, F_{up}. This upscaled map, in conjunction with the corresponding lower-level feature map from the encoder, known as F_{low}, is fed into the UAFM. Here, an α weight is calculated, quantifying the spatial relevance of each pixel. This leads to the application of an element-wise multiplication to both F_{up} and F_{low}, guided by the α values. The resultant fused image is obtained through an element-wise addition of these modified maps, a process succinctly described by Equation (1), where *Upsample*() represents the upsampling operation, *Attention*() represents the attention fusion operation, and F_{out} is the output of the fused image.

$$\begin{aligned} F_{up} &= Upsample(F_{high}) \\ \alpha &= Attention(F_{up}, F_{low}) \\ F_{out} &= F_{up} \cdot \alpha + F_{low} \cdot (1 - \alpha) \end{aligned} \tag{1}$$

Moreover, the PP-LiteSeg-T network facilitates effective contextual integration of features via the SPPM. By employing a series of global pooling operations, this module adeptly pools, convolves, and upsamples feature maps. It stands out for its streamlined design, which involves a reduction in the number of channels both at intermediate and output stages, the omission of shortcut connections, and the replacement of concatenation operations with simpler addition operations. Such modifications not only substantially reduce the computational burden but also enhance data processing speed and efficiency. These improvements ensure the network's competency in delivering real-time performance, marking a significant advancement in the field of semantic segmentation for agricultural applications.

2.2.2. Improved VINS-RGBD System

The VINS-RGBD setup builds upon the foundational VINS-Mono system [27], integrating depth data to address scale indeterminacy issues, thereby bolstering the system's overall stability. By amalgamating the VINS-RGBD architecture with the PP-LiteSeg-T network, a comprehensive system for real-time localization and 3D semantic mapping is established. This system is segmented into five primary components: measurement preprocessing, system initialization, local visual–inertial odometry (VIO), semantic segmentation, and loop detection with the mapping process, as delineated in Figure 6. The solid black boxes represent the core process modules of VINS-RGBD, while the solid red boxes represent the modules we have added.

1. Measurement Processing

Sensor-acquired data undergo a comprehensive processing phase, where they are primarily categorized into two streams: the analysis of visual data and the interpretation of inertial measurement unit (IMU) data. The visual data segment encompasses both RGB and depth imagery. In the stage of feature point extraction, the strategy employs L_SuperPoint feature points as a substitute for the traditionally utilized Shi–Tomasi feature points. The L_SuperPoint feature points are a lightweight solution based on the SuperPoint feature points, with feature extraction relying on a deep hierarchical convolutional network. The specific network architecture is described in Figure A1 of Appendix A.

Descriptors are utilized to align and monitor feature points across multiple images. This process involves the application of the K-nearest neighbor (KNN) algorithm for the precise matching of descriptors, while the random sample consensus (RANSAC) algorithm [28] efficiently filters out those feature points that fail to match. In the feature point extraction phase, images are methodically segmented into various regions. Within these regions, only the feature points demonstrating the highest rate of tracking success are preserved. This strategic selection ensures a uniform distribution of feature points throughout the image, which is crucial for enhancing the reliability and accuracy of tracking. The selection of keyframes is meticulously conducted by assessing the mean parallax observed between the current frame and the immediately preceding keyframe, coupled with an analysis of the number of feature points actively being tracked. This approach facilitates the identification of keyframes that are pivotal for maintaining the continuity and coherence of the tracking process.

In the course of handling IMU data, pre-integration is applied to the measurements from the IMUs across successive image frames. Given the higher sampling rate of the IMU compared to the rate at which image frames are captured, synchronizing IMU readings with corresponding image frames becomes essential.

2. Initialization

The initialization phase employs both pre-integrated IMU data and structure from motion (SFM) techniques. Initially, SFM deduces the three-dimensional coordinates of feature points by aligning those identified in successive RGB images and by estimating the camera's trajectory. Following this, the SFM outputs are utilized alongside IMU readings to refine estimates of the gyroscopic bias. The final step involves calculating the velocity, the direction of gravity, and the scale factor for each image frame captured by the camera.

3. Local VIO

In the VINS-RGBD system, depth information for a majority of feature points is directly sourced from depth images, with depth accuracy being enhanced through a validation process that filters out noise. For feature points located beyond the depth sensor's measurement capabilities, depth is deduced using a triangulation approach [29]. The system's back-end processes image data, feature point coordinates, and IMU data disseminated by the front-end, employing a joint optimization strategy to minimize discrepancies. This optimization relies on a marginalization operation based on the Schur complement [30] and leverages the Ceres solver for nonlinear optimization challenges [31]. As a result of this refined optimization within the sliding window framework, the system accurately ascertains both the camera's positional data and its velocity.

4. Semantic Segmentation

Keyframes received by the VINS-RGBD system's front-end undergo pixel-level semantic segmentation through the PP-LiteSeg-T network. This process identifies and visually distinguishes the objects within the images by classifying and applying unique colors to them. During the map construction phase, these semantic data are integrated into the point cloud, facilitating the classification of the point cloud and the creation of a semantic map.

5. Loop Detection and Mapping

The challenge of trajectory deviation arises due to the gradual accumulation of computational errors and noise from camera measurements. Addressing this issue necessitates the implementation of loop closure detection to refine the positioning of the camera. In tackling this pivotal task, the VINS-RGBD framework integrates the functionality of DBow2 [32], a sophisticated algorithm designed for the detection of loop closures. DBow2 commences its operation by categorizing L_SuperPoint feature points found within keyframe images into distinct clusters. This step facilitates the creation of a comprehensive visual vocabulary, effectively assembling a bag of visual words. Subsequently, the algorithm conducts a comparative analysis of the visual word bag associated with each keyframe against those

compiled from all previous keyframes. The identification of a loop closure is triggered when the comparative similarity of two keyframes exceeds a predetermined threshold. This detection initiates the execution of a global pose graph optimization process that is specifically aimed at rectifying inaccuracies in the camera's positioning.

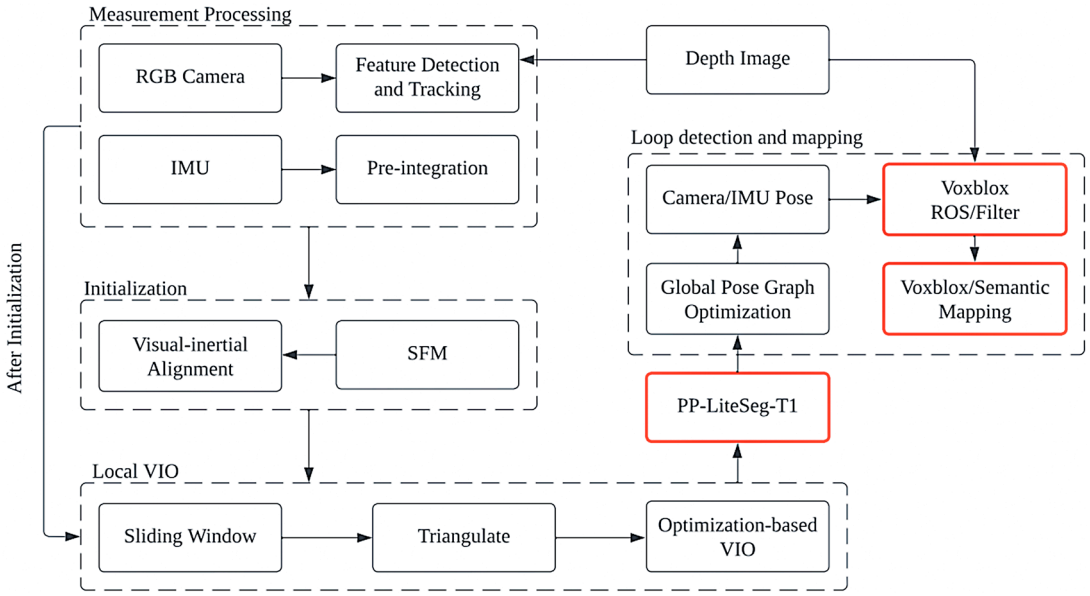


Figure 6. Improved VINS-RGBD system with integrated PP-LiteSeg-T.

2.2.3. Point Cloud Compression

1. Radius Outlier Removal Filter

Primarily due to errors in measurement and environmental obstructions such as foliage occlusion, the VINS-RGBD system's point cloud maps occasionally contain anomalies. The application of a radius filter is instrumental in enhancing the fidelity of these point cloud maps. This method effectively identifies and eliminates aberrant data points, thereby significantly improving the overall accuracy of the point cloud map.

The radius filter determines outlier points by calculating the distance between each point and its neighboring points. The specific working principle is as follows. First, for each point in the point cloud, its Euclidean distance d to all other points except itself is calculated. Then, a radius threshold r is set, any point within d<r is considered to be its neighbor, and the number n of neighboring points for each point is counted. Finally, if the number n of neighboring points for a point is less than a predefined minimum neighbor count k, then this point is removed and considered an outlier. Since outliers are sparsely distributed in dense point cloud maps, removing them does not result in significant information loss. Therefore, processing the original point cloud with this filter results in a more accurate and robust point cloud map.

Since the experiments in this paper use dense point cloud reconstruction for strawberry plants, and considering the noise effects caused by the complex data acquisition environment, the point cloud of strawberry plants is processed by setting the neighboring point radius threshold r to 0.03 and the minimum number of neighboring points k to 10.

2. Building a Voxblox Map

Due to the extensive storage requirements of point cloud maps generated by VINS-RGBD, constructing large-scale maps with restricted resources is challenging. Thus, Voxblox is employed for the representation and storage of dense point cloud maps.

The Voxblox system architecture, illustrated in Figure 7, organizes space into cubic segments, employing voxels to depict the three-dimensional environment. Primarily, Voxblox utilizes a truncated signed distance field (TSDF) for map construction and supports

the real-time incremental reconstruction of Euclidean signed distance field (ESDF) voxels and grids, leveraging the latest updates from TSDF voxels. For data storage, Voxblox adopts the voxel hashing technique [33], offering quicker data retrieval compared to the Octomap's octree-based structure [34].

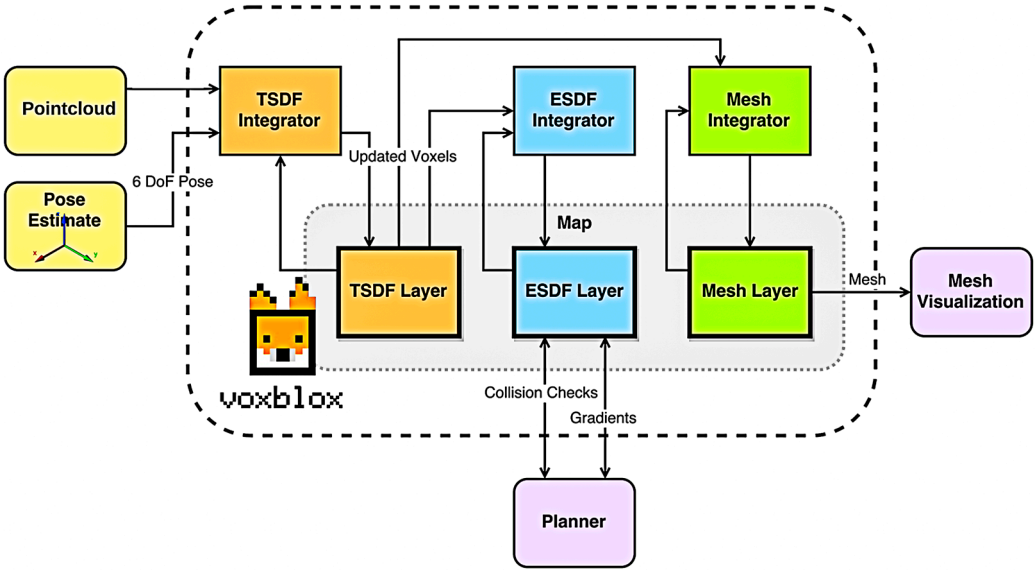


Figure 7. Voxblox system diagram.

In the construction of Voxblox, the TSDF is first developed through a procedure that utilizes a combination of weighting and merging of point cloud data. This technique ensures enhanced speed and precision in map construction, particularly with larger voxel dimensions. The integration of point cloud data into the TSDF, and, subsequently, the voxel creation, employs a specific weighting function, as depicted in Equation (2). Here, w signifies the weight assigned to a newly observed point, d is the distance from this point to the voxel's boundary surface, x denotes the voxel's spatial center, p is the position of the newly observed point, and z represents the depth captured in the image frame, while δ and ϵ, which are set at 4v and v, respectively, indicate the truncation distances, with v being the voxel size. The incorporation of new scans into the existing voxel mesh is facilitated through a method that groups ray casting, effectively enhancing the speed of ray casting without compromising accuracy. Subsequently, the ESDF is derived from the TSDF.

$$w_{quad}(x,p) = \begin{cases} \frac{1}{z^2} & -\epsilon < d \\ \frac{1}{z^2}\frac{1}{\delta-\epsilon}(d+\delta) & -\delta < d < -\epsilon \\ 0 & d < -\delta \end{cases} \quad (2)$$

3. Results

3.1. Performance Analysis of L_SuperPoint

The higher the accuracy of feature points, the better the overall mapping effect of the system, resulting in more accurate yield estimation for crops. In this study, the L_SuperPoint network was trained, and its performance was assessed using a proprietary dataset within a simulated experimental hardware setup, comprising an Intel Core i7-7800X CPU and an NVIDIA GeForce GTX 1080 Ti GPU. The deep learning framework employed was PyTorch 1.9, running on an Ubuntu 18.04 operating system.

3.1.1. Feature Point Extraction Analysis

Table 2 presents a comparison among L_SuperPoint, SuperPoint, FAST, and Harris feature points. Repeatability is defined as the proportion of matched feature point pairs relative to the overall number of features, taking into account variations in lighting or

perspective. The positioning error (PE) refers to the mean pixel distance between corresponding feature points observed from identical viewpoints. Frames per second (FPS) measures the average throughput in terms of images processed each second.

Table 2. Evaluation of feature point extraction.

	Illumination Change	Viewpoint Change	PE	FPS
L_SuperPoint	66.3%	54.7%	1.10	9.2
SuperPoint	67.8%	55.3%	1.05	2.3
FAST	60.3%	51.0%	1.93	9.5
Harris	62.5%	58.5%	1.09	1.4

The analysis results indicate that compared to the highly accurate SuperPoint, L_SuperPoint feature points exhibit a slightly lower accuracy, of 1.5%, under changes in lighting conditions and 1.4% under changes in viewpoints. Therefore, L_SuperPoint feature points have similar accuracy to SuperPoint feature points. In terms of frames per second (FPS), L_SuperPoint feature points are 6.9 FPS higher than SuperPoint feature points and 7.8 FPS higher than Harris feature points. This suggests that L_SuperPoint feature points have higher accuracy and speed during extraction.

3.1.2. Descriptor Matching Analysis

Table 3 lists the matching rate, which is the proportion of successfully matched feature points through descriptors. The data show that L_SuperPoint is only 1.2% lower than SuperPoint, about 9% higher than the ORB algorithm, and only 0.8% lower than SIFT. When evaluating the FPS metric, L_SuperPoint is only 0.4 lower than ORB, 7.0 higher than SuperPoint, and 8.3 higher than SIFT. Therefore, L_SuperPoint is an efficient feature extraction network that meets the real-time operational requirements of contemporary systems.

Table 3. Descriptor matching evaluation.

	Illumination Change	Viewpoint Change	FPS
L_SuperPoint	55.1%	51.2%	9.0
SuperPoint	56.3%	52.5%	2.0
ORB	46.7%	43.8%	9.4
SIFT	55.9%	55.7%	0.7

To assess the performance of the comprehensive 3D semantic graph reconstruction system, practical 3D reconstruction tests were carried out on several strawberry plants and rows of strawberry plants. The effectiveness of these reconstructions is thoroughly examined through quantitative analysis, with in-depth discussions presented in Section 3.3.

3.2. Semantic Segmentation Performance Analysis

To assess the PP-LiteSeg-T network's capability in semantically segmenting strawberry plants, this research inputs a dataset of strawberry plant images into the PP-LiteSeg-T for training, distributed in a 7:2:1 ratio as training, validation, and test sets, respectively. The test set specifically included both close-up and distant photographs of strawberry plants for segmentation verification, with the outcomes depicted in Figure 8. The segmentation outcomes demonstrate the model's proficiency in distinguishing the majority of strawberry fruits, branches, and leaves with clarity. Nevertheless, challenges such as lighting, shadows, and foliage occlusion result in a minority of strawberry plants and foliage being inaccurately segmented. On the whole, the PP-LiteSeg-T network effectively achieves precise segmentation of strawberry plants' branches, leaves, and fruits.

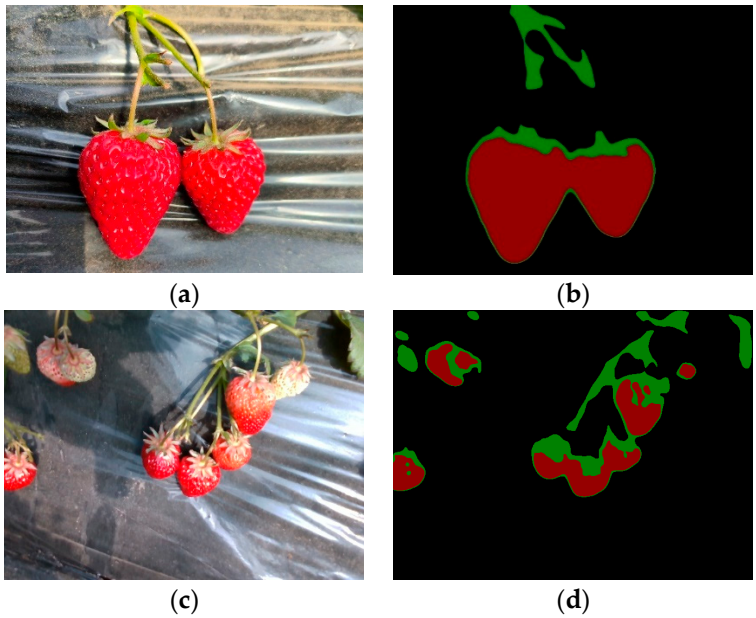


Figure 8. The semantic segmentation performance was evaluated. Below are examples of segmentation results for close-range and far-range images. In the segmentation results, red represents fruit, green represents foliage, and black represents background. (**a**) Close-up image of a strawberry plant. (**b**) Close-up segmented image. (**c**) Long-range image of strawberry plants. (**d**) Long-range segmented image. The results indicate that the PP-LiteSeg-T network can achieve effective segmentation of dataset images.

In an effort to thoroughly assess the performance of the PP-LiteSeg-T network, this study employed a self-constructed strawberry dataset to conduct a comparative analysis against several well-recognized semantic segmentation networks. The networks under comparison were optimized through TensorRT to enhance execution efficiency, setting the training duration at 100 epochs and establishing a learning rate of 0.01. The evaluation criteria selected for gauging the efficacy of these networks included the mean intersection over union (*mIoU*) and the *FPS*, with the outcomes compiled in Table 4. The *mIoU* metric, which quantifies the average proportion of overlap between the predicted and actual segments, was calculated in accordance with Equation (3), where *TP* indicates true positive outcomes, *FP* indicates false positives, and *FN* indicates false negatives. Additionally, the FPS metric, reflecting the processing speed of the semantic segmentation networks by counting the number of frames analyzed per second, was derived following Equation (4). Here, *frameNums* refers to the total count of frames evaluated, and *elapsedTime* refers to the total time used for processing. For the purpose of this evaluation, *frameNums* was fixed at 1500, and *FPS* values were determined by measuring the *elapsedTime* across various semantic segmentation networks.

$$mIoU = \frac{1}{k+1}\sum_{i=0}^{k}\frac{TP}{FN+FP+TP}\# \quad (3)$$

$$FPS = \frac{frameNums}{elapsedTime} \quad (4)$$

The data in Table 4 indicate that the average mean intersection over union (mIoU) of the PP-LiteSeg-T network is 73.2%, demonstrating good accuracy compared to other state-of-the-art networks. In terms of frames per second (FPS), the PP-LiteSeg-T network achieves a processing speed of 228.3, significantly surpassing other advanced networks. Therefore, the PP-LiteSeg-T network can provide high-precision semantic segmentation while maintaining high processing speed. It effectively meets the system's real-time segmentation and yield estimation requirements.

Table 4. Performance comparison results of classical semantic segmentation networks. Compared to other classic semantic segmentation networks, the PP-LiteSeg-T network boasts the fastest image processing speed and relatively high segmentation accuracy, achieving a balance between segmentation precision and processing speed.

Model	Encoder	mIoU (%)	FPS
ENet	-	51.8	63.1
ICNet	PSPNet50	68.2	33.9
DFANet A	Xception A	65.4	118
SwiftNet	ResNet18	71.5	-
BiSeNetV1	Xception39	66.3	178
BiSeNetV1-L	ResNet18	68.3	118.3
BiSeNetV2	-	73.7	124.5
BiSeNetV2-L	-	72.8	33.8
STDC1-Seg	STDC1	73.2	198.1
STDC2-Seg	STDC2	74.1	153.9
PP-LiteSeg-T	STDC1	73.2	**228.3**

To provide a more intuitive representation of the performance of the PP-LiteSeg-T semantic segmentation network, we evaluated the network's performance using precision–recall curves, as shown in Figure 9. The curve exhibits a stable downward trend across the entire threshold range, indicating a balance between recall and precision at various thresholds. Therefore, the semantic segmentation network demonstrates good performance.

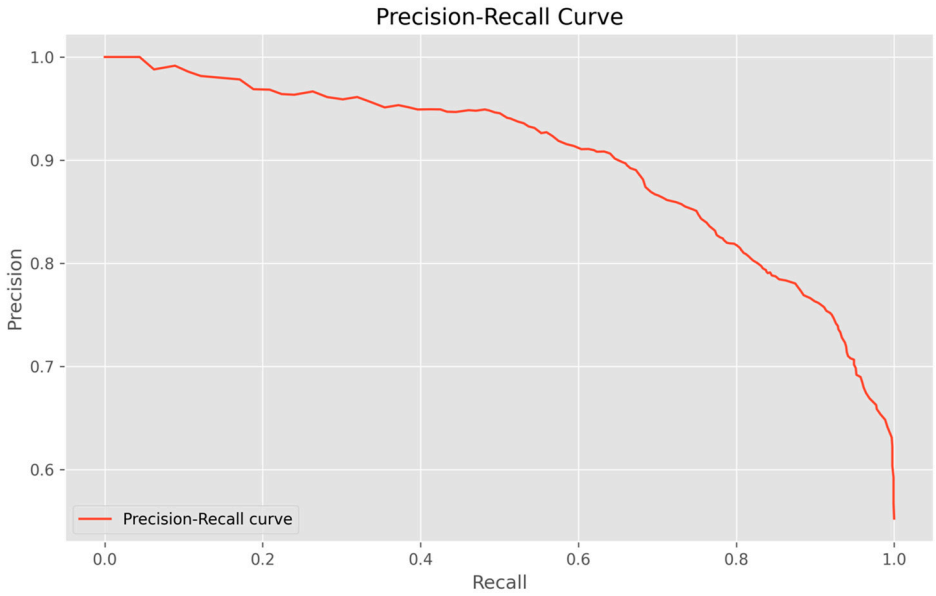


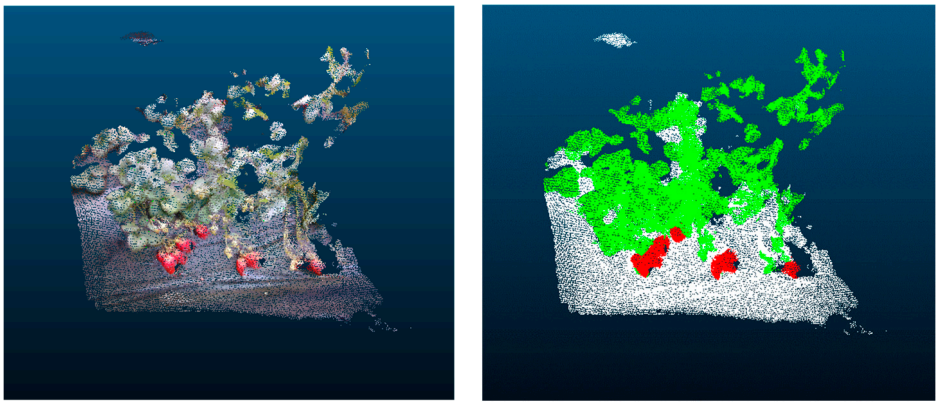
Figure 9. PP-LiteSeg-T network precision–recall curve.

3.3. Experimental Reconstruction of Strawberry Plants

3.3.1. Reconstruction Experiment of Several Strawberry Plants

Initially, this study focuses on reconstructing a 3D semantic map for several strawberry plants. In the resulting semantic point cloud, the foliage and branches of the strawberry plants are depicted in green, while the fruit is highlighted in red. The plant's background and other components are depicted in white to accentuate the display of the strawberry plant. The outcomes of this experiment are documented in Figure 10.

The experimental analysis demonstrated that the majority of the point cloud data in the reconstructed 3D model could be accurately identified. Nevertheless, due to the intricate nature of the reconstruction environment and variations in the accuracy of feature extraction and semantic segmentation, a segment of the point cloud data was misclassified.

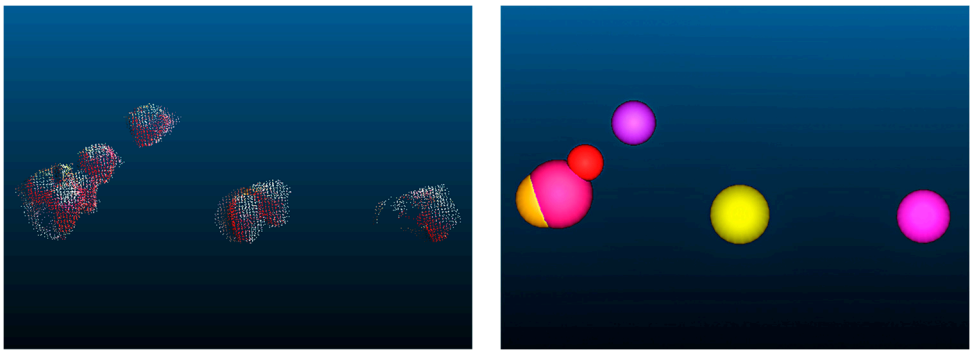


(**a**) (**b**)

Figure 10. The reconstruction results of several strawberry plants: (**a**) represents the point cloud map of the reconstruction results; (**b**) represents the semantic point cloud map of the reconstruction results.

We designed quantitative experiments to assess the yield of strawberries. Utilizing the conditional filtering algorithm [35], this study extracted point clouds representing strawberry fruits from the semantic map. The RANSAC algorithm in the CloudCompare 2.12.2 software [36] was employed for sphere fitting to the strawberry fruit point cloud. CloudCompare is a software developed by Électricité de France (EDF) for processing 3D point clouds. The culmination of this process is illustrated in Figure 11, where each sphere denotes a strawberry fruit. The precision of the 3D semantic reconstruction findings was gauged by calculating the relative error between the actual number of fruits and the predicted number, as outlined in Equation (5), where δ represents the relative error, N_t denotes the actual number of fruits, and N_p represents the predicted number of fruits. The comprehensive statistical outcomes of these evaluations are presented in Table 5.

$$\delta = \frac{|N_p - N_t|}{N_t} \times 100\% \tag{5}$$



(**a**) (**b**)

Figure 11. Using conditional filtering to extract the strawberry fruit point cloud and applying RANSAC spherical fitting for counting: (**a**) represents the extracted strawberry fruit point cloud result; (**b**) represents the result of RANSAC spherical fitting, where each sphere represents a strawberry fruit.

The relative error serves as a metric for the precision of the estimated values in comparison to the actual values, with a lower relative error signifying enhanced accuracy of the predictions. According to the data presented in Table 5, the mean relative error across several tests is recorded at 10.87%, reflecting a noticeable discrepancy between the estimated and actual counts. This deviation is attributed to various factors, including obstruction by foliage and noise within the point cloud data, which complicates the sphere fitting process undertaken by the RANSAC algorithm, leading to inaccuracies in the fruit count estimations.

Table 5. Quantitative experimental test results. The experiment evaluates the accuracy of strawberry fruit prediction using relative error. According to the table, the system can achieve relatively accurate yield estimation.

Sample	Actual Value	Predicted Value	Relative Error (%)
1	16	14	12.5
2	33	32	3.03
3	43	39	9.30
4	59	51	13.56
5	69	58	15.94
Mean	-	-	10.87

3.3.2. Reconstruction Experiment of Single-Row Strawberry Plants

1. The Original Point Cloud Map

To assess the proposed system's capability for 3D reconstruction in extensive scenes, this study constructed semantic maps for a single row of strawberry plants within a larger selection of three rows. Each row in the strawberry field spans 8 m. The resulting 3D point cloud and semantic maps are presented in Figure 12.

2. Radius Filtering Processing

In Figure 12, the detailed 3D reconstruction effectively captures the architectural complexity of an entire strawberry orchard row, accurately delineating branches, foliage, and fruits. This precision aids significantly in the estimation of fruit yields and in monitoring for pest and disease outbreaks within the orchard. Nonetheless, challenges such as variations in illumination, sensor inaccuracies, and additional disturbances introduce noise and discrepancies into the reconstructed point cloud visualization. To mitigate the influence of noise, we selected the second row for radius filtering and compared the results before and after processing. Figure 13a illustrates the effects of the point cloud map before and after radius filtering, while Figure 13b demonstrates the effects of the semantic point cloud map before and after radius filtering.

3. Voxblox Processing

Addressing the challenge of efficient 3D mapping and storage with constrained resources, the Voxblox ROS framework is utilized to convert point cloud data into Voxblox maps. These maps facilitate navigation and obstacle avoidance in robotic systems by efficiently reducing computational load and memory usage. The construction of a Voxblox map for the strawberry plants in the third row is illustrated in Figure 14.

In order to conduct yield estimation in large-scale environments, addressing the issue of map storage is crucial. We employed radius filtering and Voxblox techniques to reduce point cloud storage consumption. The experimental results regarding memory consumption for the map of three rows of strawberry plants are presented in Table 6.

The experiments demonstrate that using the radius filter can reduce memory consumption by an average of 5.59%, while employing Voxblox maps can reduce memory consumption by an average of 96.91%. This indicates that the proposed system can efficiently perform large-scale 3D mapping tasks to support yield estimation for crops in large environments.

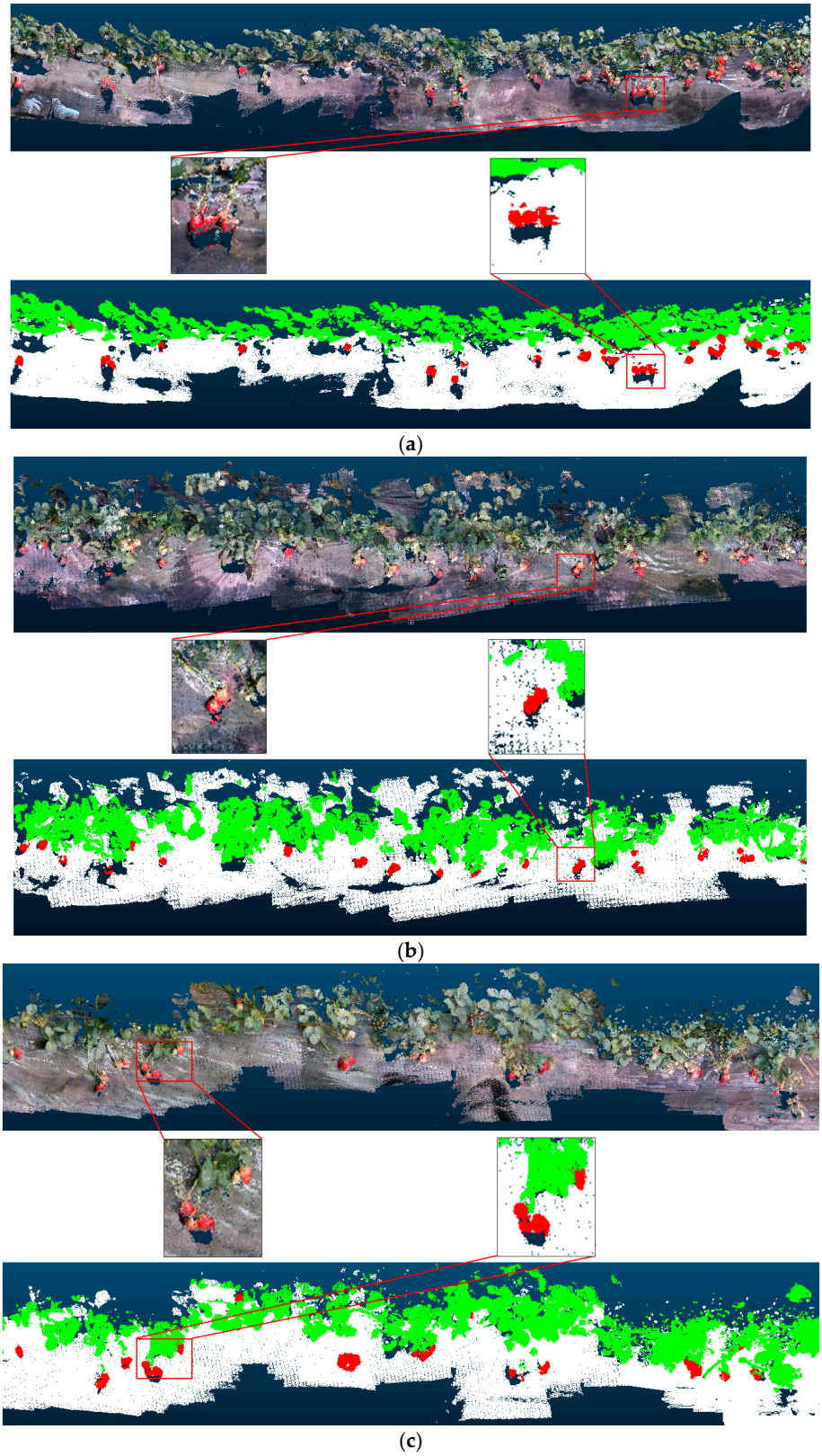
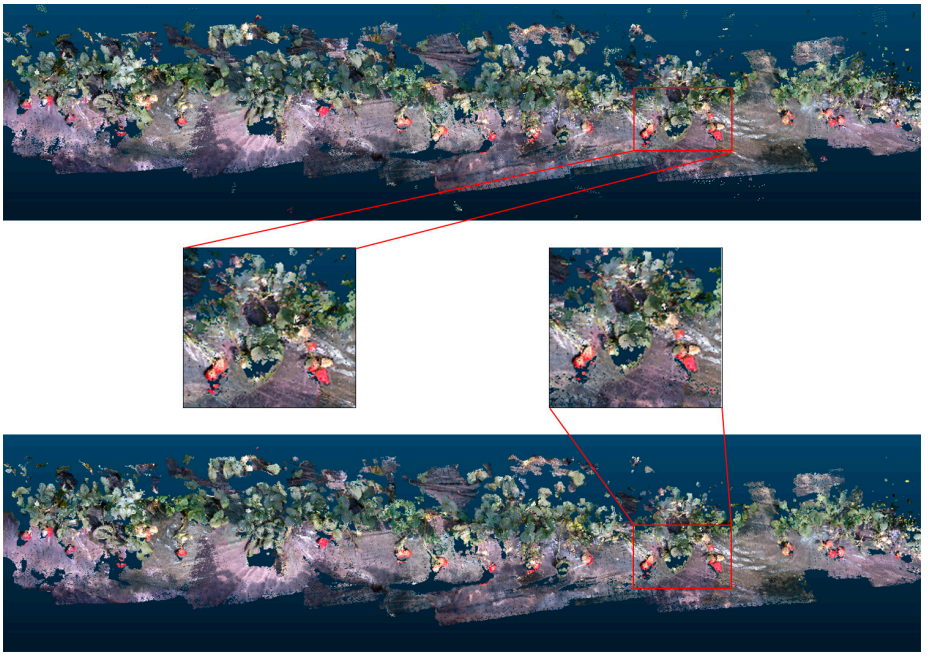
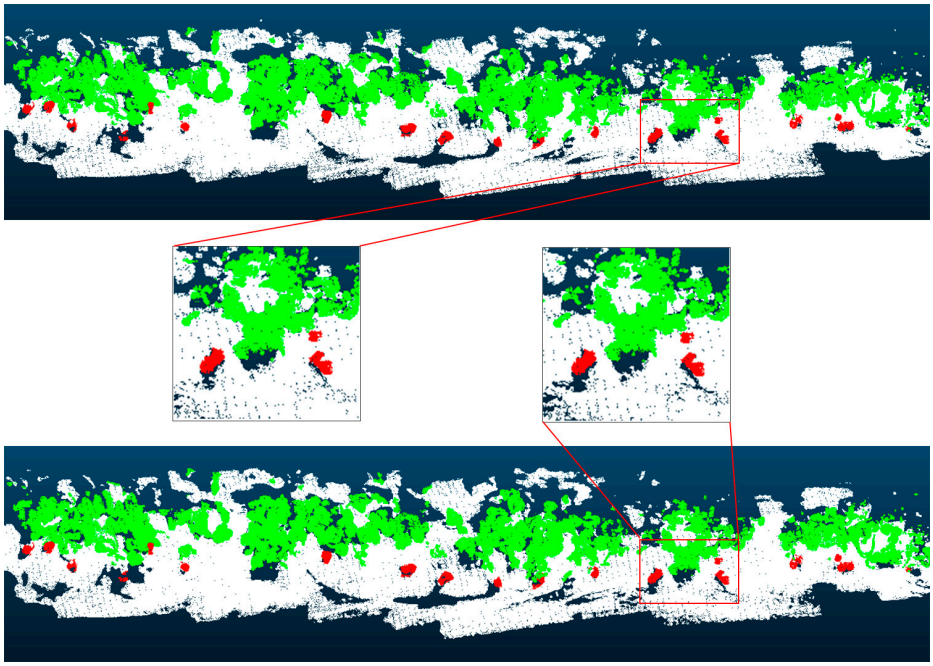


Figure 12. The experimental results of entire-row strawberry plant reconstruction; each row includes a point cloud map and a semantic point cloud map. The following are the reconstruction results of three rows of strawberry plants. (**a**) The 3D reconstruction results of the first row. (**b**) The 3D reconstruction results of the second row. (**c**) The 3D reconstruction results of the third row. In the semantic point cloud diagram, red represents the fruit, green represents the branches and leaves, and white represents the background.



(**a**)



(**b**)

Figure 13. The reconstruction results of the second row filtered using radius filtering. (**a**) represents the point cloud maps before and after radius filtering. (**b**) represents the semantic point cloud maps before and after radius filtering.

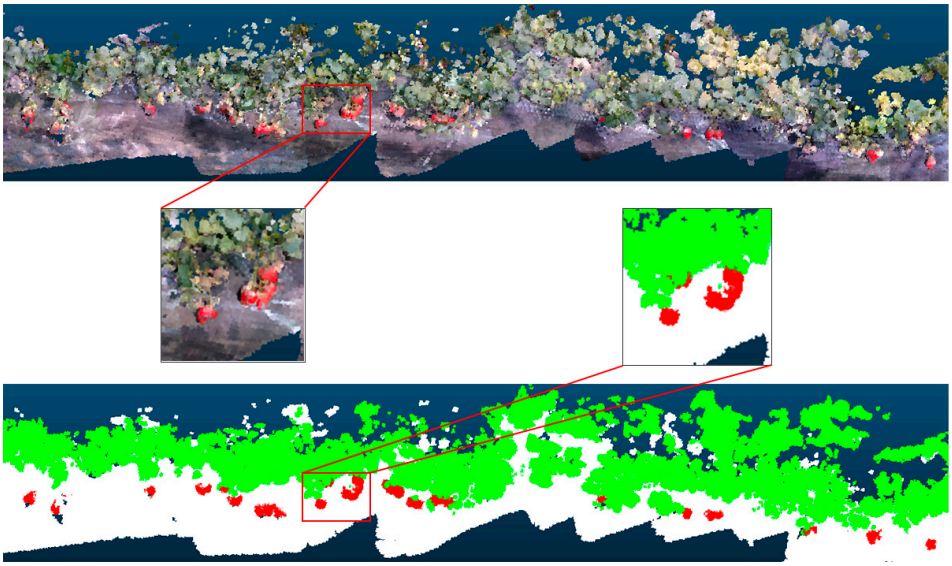


Figure 14. Voxblox mapping results. In the semantic point cloud diagram, red represents the fruit, green represents the branches and leaves, and white represents the background.

Table 6. Comparison of memory consumption of different maps. Using radius filtering and Voxblox maps can significantly reduce map storage consumption.

Dataset	Semantic Map (MB)	After Filtering (MB)	Voxblox (MB)	Memory Savings after Filtering (%)	Voxblox Memory Savings (%)
First row	93.6	89.3	3.3	4.59	96.47
Second row	103.3	100.2	3.4	3.01	96.71
Third row	152.8	142.5	4.2	6.74	97.25
First and second rows	203.5	189.5	6.3	6.88	96.90
Second and third rows	268.1	249.3	8.4	6.98	96.87
Three-row	354.7	335.6	9.8	5.38	97.23
mean	-	-	-	5.59	96.91

3.4. Real-Time Detection Experiments

To guarantee that the map-building process is both rapid and effective, it is crucial to assess the system's real-time capabilities. Within the enhanced VINS-RGBD framework, the necessity for real-time operations is divided across its components: the front-end module, tasked with feature extraction and tracking, alongside the semantic segmentation module, must operate in real time. In contrast, the back-end module, responsible for optimization and loop closure detection, does not share this requirement. Consequently, the overall real-time performance hinges on the collective processing velocity of the modules mandated to function in real time. Evaluations of these critical real-time modules were conducted on a hardware setup equipped with an Intel Core i7-7800X CPU and an NVIDIA GeForce GTX 1080 Ti GPU.

Table 7 delineates the performance metrics, revealing that the feature extraction and tracking component processes each video frame in an average of 13.61 ms. Concurrently, the semantic segmentation component exhibits an even more efficient average processing time of 8.67 ms per frame. Considering the operational frame rate of 30 frames per second for video data captured by the D435i camera, these processing durations affirm that the system successfully adheres to the requisites for real-time performance.

Table 7. Real-time processing results per frame.

Module	Thread	Time (ms)
VINS-RGBD	Feature detection and tracking	13.61
PP-LiteSeg-T	Semantic segmentation	8.67

3.5. Localization Accuracy Evaluation Experiments

Localization accuracy is a crucial metric for assessing the stability of SLAM systems, as it directly impacts the quality of the final map. More accurate yield estimation relies on high-quality maps. Therefore, the localization accuracy of the proposed system is evaluated.

The focus of this study was to assess the robustness of the VINS-RGBD system's utilization of L_SuperPoint feature points, in comparison to the conventional Shi–Tomasi corner detection approach, through a methodical evaluation using two distinct datasets. The inaugural dataset was acquired via a hand-held device during the Tsukuba Challenge 2022, and the subsequent dataset was similarly obtained at the Ikuta Campus of Meiji University, employing the same data collection methodology. Illustrative of the datasets' reference trajectories, Figure 15 provides GPS-based trajectory mappings. To quantitatively measure the system's performance, this study employed *RPE* and absolute trajectory error (ATE) as the principal evaluation metrics. While *ATE* offers an overarching view of the trajectory deviations, *RPE* zeroes in on pose changes, making it particularly useful for assessing the extent of trajectory drift.

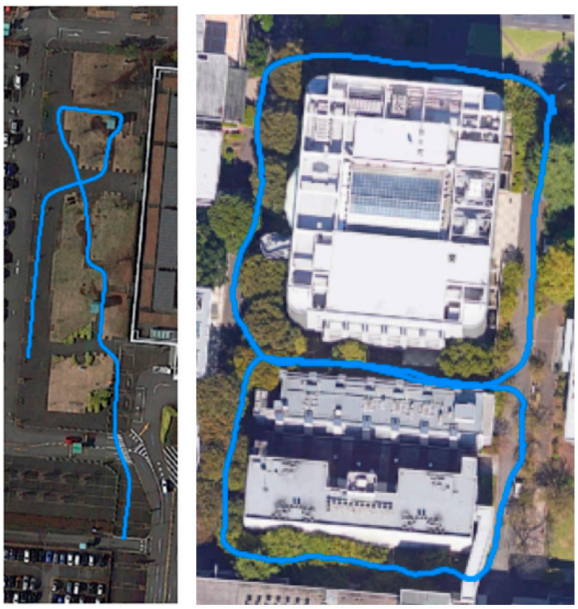


Figure 15. Measuring reference trajectories.

Relative pose error (*RPE*) quantifies the divergence between the estimated positions by the system and the actual positions across two frames within a predetermined time span, Δ. This evaluation assumes the system's estimated trajectory, denoted as $P_1, \ldots, P_n \in SE(3)$, and the actual trajectory, represented as $Q_1, \ldots, Q_n \in SE(3)$, with each subscript marking the timestamp. *RPE* is calculated as per Equation (6), where E_i signifies the relative positional error at the ith timestamp, $trans(X)$ is the translational component of any element $X \in SE(3)$, and n symbolizes the total count of frames evaluated:

$$RPE(i) = \sqrt{\frac{1}{n}\|trans(E_i)\|}$$

$$E_i = (Q^{-1}Q_{i+\Delta})^{-1}(P_i^{-1}P_{i+\Delta}) \quad (6)$$

ATE measures the overall discrepancy between the system's estimated trajectory and the actual path. As delineated in Equation (7), F_i represents the magnitude of trajectory deviation at each timestep, with S embodying the transformation matrix that aligns the estimated trajectory S_i to the actual trajectory Q_i.

$$ATE(i) = \sqrt{\frac{1}{n}\|trans(F_i)\|}$$

$$F_i = Q_i^{-1} S P_i \tag{7}$$

Table 8 provides the *ATE* and *RPE* metrics in meters for the VINS-RGBD system, comparing the performance of Shi–Tomasi corner points against L_SuperPoint feature points. To ensure uniformity across both datasets, scale adjustments were applied using the Umeyama method, with calculations performed through evo.

Table 8. Localization accuracy analysis of different feature points of VINS-RGBD. L_SuperPoint feature points achieved lower *ATE* and *RPE*, resulting in higher localization accuracy.

Method	No. 1 Dataset		No. 2 Dataset		Mean	
	ATE	*RPE*	*ATE*	*RPE*	*ATE*	*RPE*
Shi–Tomasi	5.164	0.267	4.456	0.204	4.810	0.236
L_SuperPoint	3.512	0.241	2.241	0.146	2.877	0.194

Table 8 demonstrates that the average *ATE* of L_SuperPoint feature points on both datasets is 1.933 lower than that of Shi–Tomasi, and the average RPE is lower by 0.042. This indicates that using L_SuperPoint feature points within the VINS-RGBD framework leads to improved localization accuracy.

Figure 16 shows the results of comparing the experimental trajectories of the VINS-RGBD system when using the two feature points.

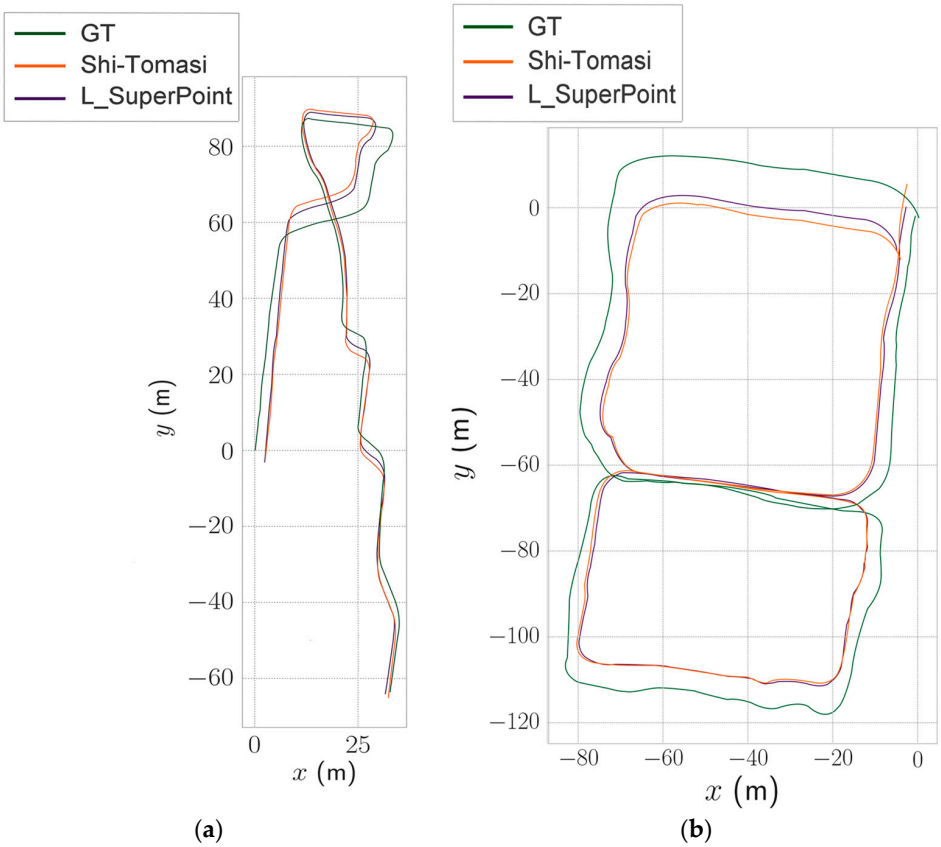


Figure 16. Localization accuracy experiment results, where green represents the ground truth trajectory, orange represents the trajectory when the system uses Shi–Tomasi feature points, and purple represents the trajectory when the system uses L_SuperPoint feature points. (**a**) represents the experimental results on the Tsukuba Challenge 2022 dataset. (**b**) represents the experimental results on the Meiji University Ikuta Campus dataset.

4. Discussion

Performing real-time 3D semantic reconstruction and yield estimation of strawberry plants is an exploratory endeavor, which comes with certain limitations during experimentation.

1. Position accuracy is a key metric for evaluating SLAM performance. In Section 3.5, we evaluated the positioning accuracy of our proposed system using two publicly available datasets. Public datasets typically utilize high-precision equipment or motion capture devices to obtain ground truth, which can be costly. Due to limitations in our experimental setup, we were unable to acquire accurate ground truth; hence, we opted to validate it using publicly available datasets. From Table 8, it can be observed that although the proposed system utilizes higher-precision feature points, the estimated trajectory still exhibits a certain drift compared to the actual trajectory. Apart from the influence of the algorithm itself, the positioning accuracy of SLAM is also affected by factors such as environmental complexity and camera resolution. Therefore, in future work, complementary high-precision devices, such as LiDAR sensors, RTK, GPS, etc., can be integrated to provide more accurate information to the SLAM system, thereby compensating for the inaccuracies in single-sensor state estimation.
2. The accuracy of semantic segmentation has a significant impact on establishing a three-dimensional semantic point cloud map. In this system, the PP-LiteSeg-T semantic segmentation network achieved an mIoU of 73.2% and an FPS of 228.3. Compared to other semantic segmentation networks, it demonstrates strong superiority in balancing segmentation accuracy and processing speed. Despite the PP-LiteSeg-T network's superior performance in terms of speed and accuracy over conventional semantic segmentation frameworks, it encounters certain limitations. As evidenced in Figure 17, challenges such as mislabeling, obstruction by leaves, fluctuating light conditions, and sensor inaccuracies impede the network's capability to precisely interpret semantic details. These issues contribute to inaccuracies in semantic segmentation, resulting in misclassified segments. To advance the efficacy of the system, future initiatives could include curating datasets with more pronounced features, enhancing the semantic segmentation algorithm, and integrating sensors with enhanced accuracy. Implementing these refinements is expected to significantly improve the quality of semantic segmentation results. Although PP-LiteSeg-T demonstrated excellent semantic segmentation performance on our self-constructed dataset, overfitting still occurs when deploying our model to perform segmentation tasks on other similar datasets. This is mainly due to factors such as time constraints and device conditions, which limited the amount of strawberry image data used for training the PP-LiteSeg-T network. Consequently, the model's learning of image characteristics was not comprehensive enough, leading to overfitting. To mitigate the impact of overfitting, it is necessary to increase the number of samples in the dataset.
3. This study's main findings were as follows. (1) Feature point performance analysis: Experiments were conducted on substituted feature points aiming to enhance the accuracy of image matching using higher-precision feature points. The results indicate that, compared to other traditional feature points, L_SuperPoint feature points exhibit higher accuracy and faster extraction speed. (2) Semantic segmentation performance: Performance analysis of the PP-LiteSeg-T semantic segmentation network was carried out. It demonstrated higher segmentation accuracy and faster segmentation speed compared to other traditional semantic segmentation networks. (3) Yield estimation capability: Experiments on three-dimensional semantic reconstruction of strawberries were designed. The experiments involved extracting strawberry fruits and estimating their yield. The average relative error in predicting fruit quantity compared to actual fruit quantity was only 10.87%. Additionally, modeling of multiple rows of strawberries was conducted to assess the effectiveness of Voxblox map storage. The experiments showed that using Voxblox could reduce memory consumption by an average of 96.61%, meeting the demand for outdoor crop modeling and yield estimation. (4) Real-time performance and localization accuracy: Evaluation of the

real-time performance and localization accuracy of the overall system was conducted. The experiments demonstrated that the system could operate in real time with good localization accuracy. The research experiments mentioned above indicate that the proposed system can achieve the goal of real-time yield estimation for outdoor crops.

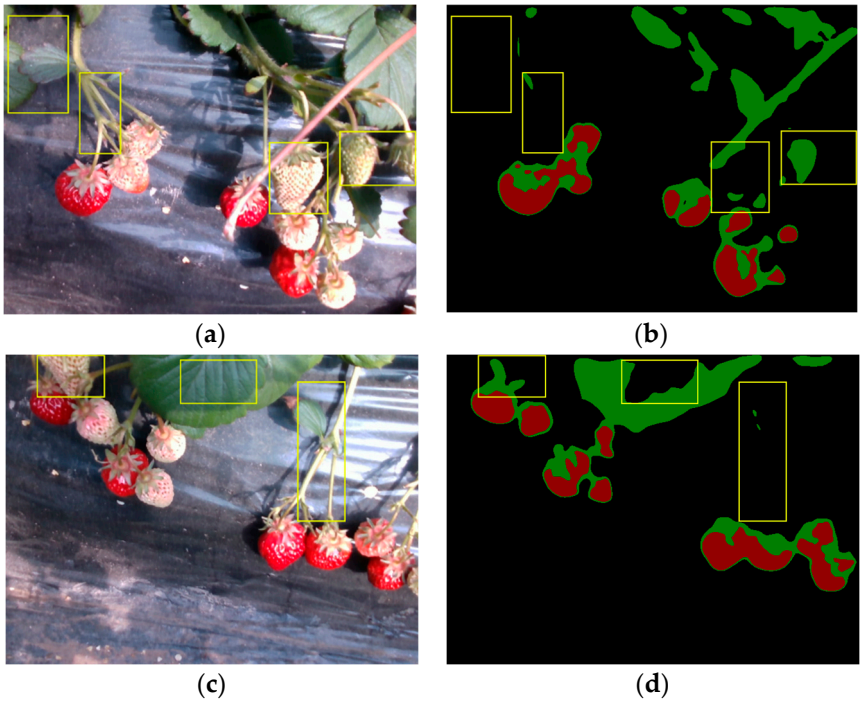


Figure 17. Due to factors such as leaf occlusion and varying light intensity, semantic segmentation may yield misclassified results. (**a**,**c**) represent the original strawberry images, while (**b**,**d**) represent the misclassified results, with areas of misclassification highlighted by yellow boxes.

5. Conclusions

This study proposes an enhanced VINS-RGBD system that integrates a semantic segmentation module to achieve the three-dimensional semantic reconstruction of strawberry plants and estimates the yield of strawberry fruits.

The system utilizes lightweight semantic segmentation networks for the real-time segmentation of image frames and effectively addresses the challenge of constructing real-time three-dimensional semantic point cloud maps by mapping semantic information onto point clouds. This facilitates real-time yield estimation. The system employs radius filtering and Voxblox technology to compress and store point cloud data, addressing the challenge of large-scale mapping in SLAM systems. This facilitates large-scale map construction and yield estimation in orchards.

The experimental findings indicate that the enhanced system achieves a semantic segmentation accuracy (mIoU) of 73.2%, a relative error in strawberry fruit quantity prediction of 10.87%, and a significant reduction in memory usage (by an average of 96.61%) through the utilization of Voxblox for point cloud storage and representation. The system's front-end feature detection and tracking exhibit an average processing time of 13.61 ms per image frame, and the semantic segmentation network processes at an average of 8.67 ms per image frame, demonstrating the system's robust real-time performance.

Currently, the main challenge in constructing real-time three-dimensional semantic maps is how to improve semantic segmentation accuracy and localization accuracy in complex environments while balancing system accuracy and real-time performance. In the future, this can be achieved through methods such as sensor fusion [37,38], optimizing poses using NERF [39,40], and using devices with higher accuracy to improve the quality of constructing three-dimensional semantic maps, ultimately aiming for more accurate

yield estimation. The project also plans to deploy devices on unmanned vehicles to achieve autonomous navigation, mapping, and yield estimation.

Author Contributions: Conceptualization, W.L.; methodology, Q.Y.; writing—original draft preparation, P.W.; writing—review and editing, Y.Z.; visualization, H.C.; supervision, Z.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Central Guidance on Local Science and Technology Development Fund of Hebei Province with grant number 226Z0302G&236Z7201G.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors upon request.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

1. The three key modules of the PP-LiteSeg-T network

Figure A1 illustrates the Flexible Lightweight Decoder (FLD) module, Figure A2 depicts the Unified Attention Fusion Module (UAFM), and Figure A3 showcases the Simple Pyramid Pooling Module (SPPM). Detailed descriptions of these modules can be found in Section 2.2.1.

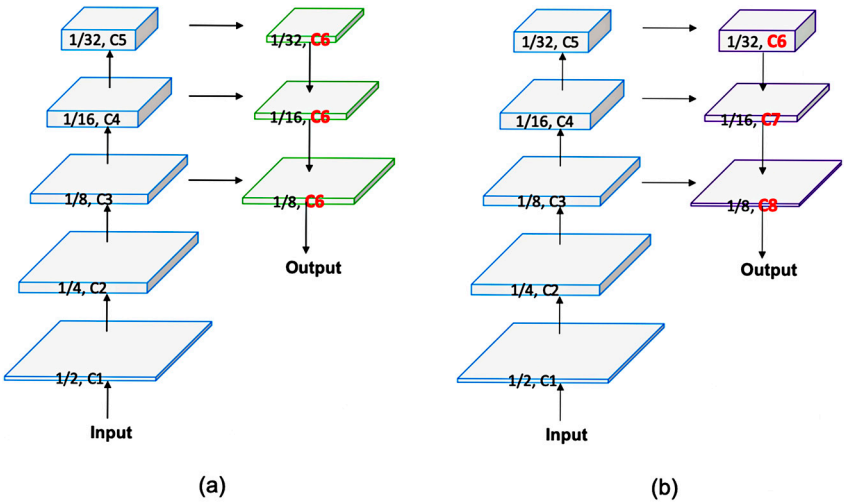


Figure A1. The Flexible Lightweight Decoder (FLD) module. (**a**) denotes the conventional decoder module, and (**b**) denotes The Flexible Lightweight Decoder (FLD) module.

2. The L_SuperPoint network architecture

The L_SuperPoint feature point represents a streamlined adaptation of the SuperPoint feature point, leveraging a depth-wise hierarchical convolutional network for feature extraction. This network is architecturally partitioned into five distinct layers. The initial layer employs a conventional convolutional network to comprehensively extract features from the raw image. Subsequently, the remaining four layers utilize a depth-wise separable convolutional network aiming to simultaneously extract pertinent information while minimizing computational demands. The detailed architecture of this network is depicted in Figure A4. Within the depth-wise separable convolutional layers, to mitigate the substantial information loss associated with downsampling, the activation function in the depth-wise convolutional layer is modified from the standard ReLU to the ReLU6 function. Meanwhile, in the point-wise convolutional layer, the ReLU function is substituted with a linear function to preserve more detail.

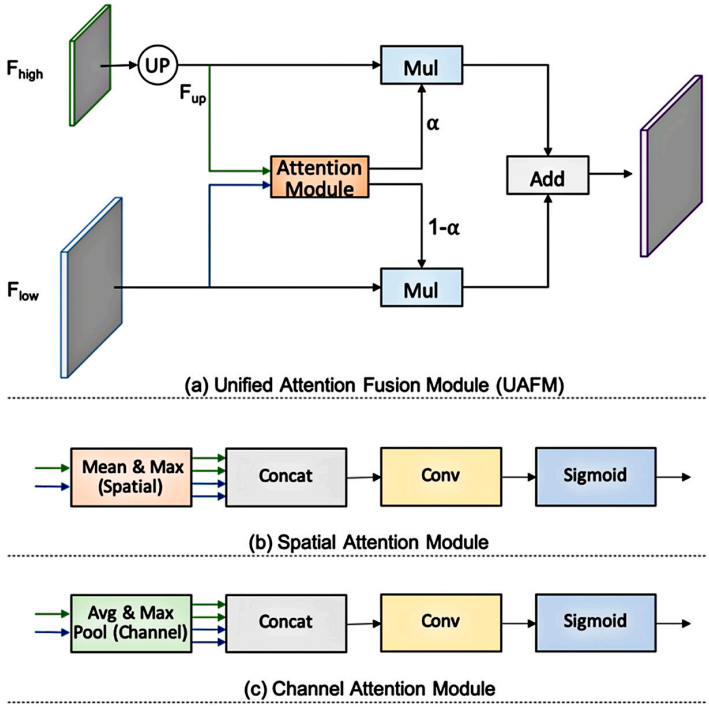


Figure A2. The Unified Attention Fusion Module (UAFM).

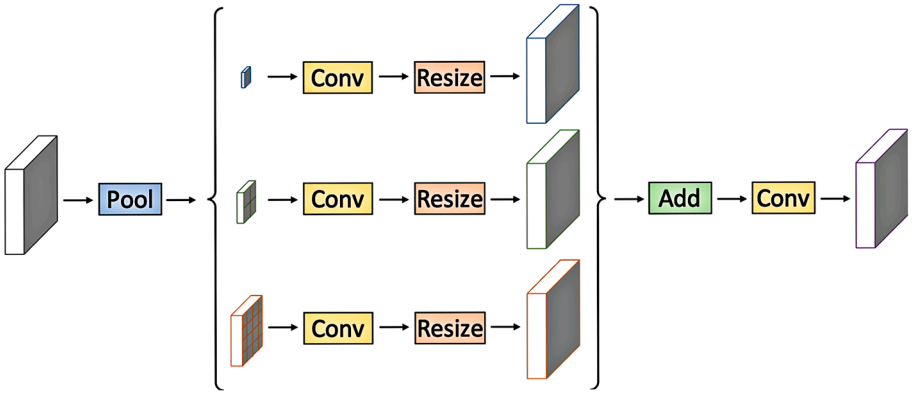


Figure A3. The Simple Pyramid Pooling Module (SPPM).

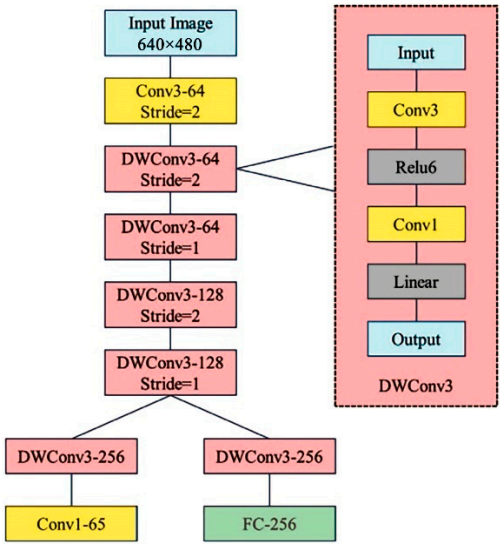


Figure A4. L_SuperPoint network architecture.

References

1. Zhang, C.; Valente, J.; Kooistra, L.; Guo, L.; Wang, W. Orchard management with small unmanned aerial vehicles: A survey of sensing and analysis approaches. *Precis. Agric.* **2021**, *22*, 2007–2052. [CrossRef]
2. Shaikh, T.A.; Rasool, T.; Lone, F.R. Towards leveraging the role of machine learning and artificial intelligence in precision agriculture and smart farming. *Comput. Electron. Agric.* **2022**, *198*, 107119. [CrossRef]
3. Xiong, J.; Liang, J.; Zhuang, Y.; Hong, D.; Zheng, Z.; Liao, S.; Hu, W.; Yang, Z. Real-time localization and 3D semantic map reconstruction for unstructured citrus orchards. *Comput. Electron. Agric.* **2023**, *213*, 108217. [CrossRef]
4. Torres-Sánchez, J.; López-Granados, F.; Borra-Serrano, I.; Manuel Peña, J. Assessing UAV-collected image overlap influence on computation time and digital surface model accuracy in olive orchards. *Precis. Agric.* **2018**, *19*, 115–133. [CrossRef]
5. Xue, J.; Fan, Y.; Su, B.; Fuentes, S. Assessment of canopy vigor information from kiwifruit plants based on a digital surface model from unmanned aerial vehicle imagery. *Int. J. Agric. Biol. Eng.* **2019**, *12*, 165–171. [CrossRef]
6. Sabzi, S.; Abbaspour-Gilandeh, Y.; García-Mateos, G.; Ruiz-Canales, A.; Molina-Martínez, J.M. Segmentation of apples in aerial images under sixteen different lighting conditions using color and texture for optimal irrigation. *Water* **2018**, *10*, 1634. [CrossRef]
7. Wang, P.; Hafshejani, B.A.; Wang, D. An improved multilayer perceptron approach for detecting sugarcane yield production in IoT based smart agriculture. *Microprocess. Microsyst.* **2021**, *82*, 103822. [CrossRef]
8. Li, J.; Tang, Y.; Zou, X.; Lin, G.; Wang, H. Detection of fruit-bearing branches and localization of litchi clusters for vision-based harvesting robots. *IEEE Access* **2020**, *8*, 117746–117758. [CrossRef]
9. Montoya-Cavero, L.-E.; Torres, R.D.d.L.; Gómez-Espinosa, A.; Cabello, J.A.E. Vision systems for harvesting robots: Produce detection and localization. *Comput. Electron. Agric.* **2022**, *192*, 106562. [CrossRef]
10. Li, Z.; Wang, J.; Zhang, Z.; Jin, F.; Yang, J.; Sun, W.; Cao, Y. A Method Based on Improved iForest for Trunk Extraction and Denoising of Individual Street Trees. *Remote Sens.* **2022**, *15*, 115. [CrossRef]
11. Sukvichai, K.; Noppanut, T.; Kan, Y. Implementation of a Monocular ORB SLAM for an Indoor Agricultural Drone. In Proceedings of the 2023 Third International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP), Bangkok, Thailand, 18–20 January 2023; IEEE: New York, NY, USA, 2023.
12. Ramirez, G.; de Oca, A.M.; Flores, G. 3D maps of vegetation indices generated onboard a precision agriculture UAV. In Proceedings of the 2023 International Conference on Unmanned Aircraft Systems (ICUAS), Warsaw, Poland, 6–9 June 2023; IEEE: New York, NY, USA, 2023.
13. Gimenez, J.; Sansoni, S.; Tosetti, S.; Capraro, F.; Carelli, R. Trunk detection in tree crops using RGB-D images for structure-based ICM-SLAM. *Comput. Electron. Agric.* **2022**, *199*, 107099. [CrossRef]
14. Mitrofanova, O.; Blekanov, I.; Sevostyanov, D.; Zhang, J.; Mitrofanov, E. Development of a Robot for Agricultural Field Scouting. In *International Conference on Interactive Collaborative Robotics*; Springer Nature: Cham, Switzerland, 2023.
15. Meyer, L.; Gedschold, J.; Wegner, T.E.; Del Galdo, G.; Kalisz, A. Enhancement of Vision-Based 3D Reconstruction Systems Using Radar for Smart Farming. In Proceedings of the 2022 IEEE Workshop on Metrology for Agriculture and Forestry (MetroAgriFor), Perugia, Italy, 3–5 November 2022; IEEE: New York, NY, USA, 2022.
16. Yuan, J.; Hong, J.; Sattar, J.; Isler, V. ROW-SLAM: Under-canopy cornfield semantic SLAM. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 23–27 May 2022; IEEE: New York, NY, USA, 2022.
17. Pan, Y.; Cao, H.; Hu, K.; Kang, H.; Wang, X. A Novel Mapping and Navigation Framework for Robot Autonomy in Orchards. *arXiv* **2023**, arXiv:2308.16748.
18. Wei, S.; Wang, S.; Li, H.; Liu, G.; Yang, T.; Liu, C. A Semantic Information-Based Optimized vSLAM in Indoor Dynamic Environments. *Appl. Sci.* **2023**, *13*, 8790. [CrossRef]
19. Dong, N.; Chi, R.; Zhang, W. LiDAR Odometry and Map Based on Semantic Information for Maize Field. *Agronomy* **2022**, *12*, 3107. [CrossRef]
20. Liu, T.; Chopra, N.; Samtani, J. Information System for Detecting Strawberry Fruit Locations and Ripeness Conditions in a Farm. *Biol. Life Sci. Forum* **2022**, *16*, 22. [CrossRef]
21. Yan, Y.; Zhang, B.; Zhou, J.; Zhang, Y.; Liu, X.A. Real-Time Localization and Mapping Utilizing Multi-Sensor Fusion and Visual–IMU–Wheel Odometry for Agricultural Robots in Unstructured, Dynamic and GPS-Denied Greenhouse Environments. *Agronomy* **2022**, *12*, 1740. [CrossRef]
22. Islam, R.; Habibullah, H.; Hossain, T. AGRI-SLAM: A real-time stereo visual SLAM for agricultural environment. *Auton. Robot.* **2023**, *47*, 649–668. [CrossRef]
23. Li, Q.; Wang, X.; Wu, T.; Yang, H. Point-line feature fusion based field real-time RGB-D SLAM. *Comput. Graph.* **2022**, *107*, 10–19. [CrossRef]
24. Zhang, Y.; Sun, H.; Zhang, F.; Zhang, B.; Tao, S.; Li, H.; Qi, K.; Zhang, S.; Ninomiya, S.; Mu, Y. Real-Time Localization and Colorful Three-Dimensional Mapping of Orchards Based on Multi-Sensor Fusion Using Extended Kalman Filter. *Agronomy* **2023**, *13*, 2158. [CrossRef]
25. Liu, T.; Kang, H.; Chen, C. ORB-Livox: A real-time dynamic system for fruit detection and localization. *Comput. Electron. Agric.* **2023**, *209*, 107834. [CrossRef]
26. Torralba, A.; Russell, B.C.; Yuen, J. Russell, and Jenny Yuen. Labelme: Online image annotation and applications. *Proc. IEEE* **2010**, *98*, 1467–1484. [CrossRef]

27. Qin, T.; Li, P.; Shen, S. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020. [CrossRef]
28. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*; Cambridge University Press: Cambridge, UK, 2003.
29. Hartley, R.I.; Peter, S. Triangulation. *Comput. Vis. Image Underst.* **1997**, *68*, 146–157. [CrossRef]
30. Sibley, G.; Matthies, L.; Sukhatme, G. Sliding window filter with application to planetary landing. *J. Field Robot.* **2010**, *27*, 587–608. [CrossRef]
31. Agarwal, S.; Mierle, K. *Ceres Solver: Tutorial & Reference*; Google Inc.: Mountain View, CA, USA, 2012; Volume 2, p. 8.
32. Galvez-López, D.; Tardos, J.D. Bags of binary words for fast place recognition in image sequences. *IEEE Trans. Robot.* **2012**, *28*, 1188–1197. [CrossRef]
33. Nießner, M.; Zollhöfer, M.; Izadi, S.; Stamminger, M. Real-time 3D reconstruction at scale using voxel hashing. *ACM Trans. Graph. ToG* **2013**, *32*, 1–11. [CrossRef]
34. Oleynikova, H.; Taylor, Z.; Fehr, M.; Siegwart, R.; Nieto, J. Voxblox: Incremental 3d euclidean signed distance fields for on-board mav planning. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; IEEE: New York, NY, USA, 2017.
35. Rusu, R.B.; Steve, C. 3d is here: Point cloud library (pcl). In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; IEEE: New York, NY, USA, 2011.
36. Girardeau-Montaut, D. CloudCompare. *Fr. EDF RD Telecom ParisTech* **2016**, *11*.
37. Cai, Y.; Ou, Y.; Qin, T. Improving SLAM techniques with integrated multi-sensor fusion for 3D reconstruction. *Sensors* **2024**, *24*, 2033. [CrossRef]
38. Feng, C.-Q.; Li, B.-L.; Liu, Y.-F.; Zhang, F.; Yue, Y.; Fan, J.-S. Crack assessment using multi-sensor fusion simultaneous localization and mapping (SLAM) and image super-resolution for bridge inspection. *Autom. Constr.* **2023**, *155*, 105047. [CrossRef]
39. Katragadda, S.; Lee, W.; Peng, Y.; Geneva, P.; Chen, C.; Guo, C.; Li, M.; Huang, G. NeRF-VINS: A Real-time Neural Radiance Field Map-based Visual-Inertial Navigation System. *arXiv* **2023**, arXiv:2309.09295.
40. Liu, J.; Nie, Q.; Liu, Y.; Wang, C. Nerf-loc: Visual localization with conditional neural radiance field. In Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA), London, UK, 29 May–2 June 2023; IEEE: New York, NY, USA, 2023.