sensors

MDPI

Article

Stereo and LiDAR Loosely Coupled SLAM Constrained Ground Detection

Tian Sun [1], Lei Cheng [2], Ting Zhang [3], Xiaoping Yuan [4], Yanzheng Zhao [1] and Yong Liu [2,*]

[1] School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China; jiaodat@sjtu.edu.cn (T.S.); yzh-zhao@sjtu.edu.cn (Y.Z.)
[2] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China; 18851792895@163.com
[3] School of Computer and Electronic Information, Nanjing Normal University, Nanjing 210023, China; zting135@nnu.edu.cn
[4] School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221116, China; xpyuankd@163.com
* Correspondence: liuy1602@njust.edu.cn

Abstract: In many robotic applications, creating a map is crucial, and 3D maps provide a method for estimating the positions of other objects or obstacles. Most of the previous research processes 3D point clouds through projection-based or voxel-based models, but both approaches have certain limitations. This paper proposes a hybrid localization and mapping method using stereo vision and LiDAR. Unlike the traditional single-sensor systems, we construct a pose optimization model by matching ground information between LiDAR maps and visual images. We use stereo vision to extract ground information and fuse it with LiDAR tensor voting data to establish coplanarity constraints. Pose optimization is achieved through a graph-based optimization algorithm and a local window optimization method. The proposed method is evaluated using the KITTI dataset and compared against the ORB-SLAM3, F-LOAM, LOAM, and LeGO-LOAM methods. Additionally, we generate 3D point cloud maps for the corresponding sequences and high-definition point cloud maps of the streets in sequence 00. The experimental results demonstrate significant improvements in trajectory accuracy and robustness, enabling the construction of clear, dense 3D maps.

Keywords: pose optimization; simultaneous localization and mapping; point cloud; 3D reconstruction; sensor fusion

check for updates

Citation: Sun, T.; Cheng, L.; Zhang, T.; Yuan, X.; Zhao, Y.; Liu, Y. Stereo and LiDAR Loosely Coupled SLAM Constrained Ground Detection. *Sensors* **2024**, *24*, 6828. https://doi.org/10.3390/s24216828

Academic Editor: Udo Frese

Received: 6 June 2024
Revised: 27 September 2024
Accepted: 18 October 2024
Published: 24 October 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Accurate localization is imperative for the effective operation of autonomous vehicles. The previous research has explored various sensor options for localization in complex environments. Although LiDAR is widely regarded as a reliable sensor for mapping and localization due to its precise range measurements, its high cost and inability to extract object colors pose challenges for widespread applications. Conversely, the GPS excels in signal-rich areas but falters in accurately localizing vehicles within urban and indoor environments. Cameras have emerged as a cost-effective alternative to LiDAR, boasting advantages such as affordability, compact size, and the ability to capture color information. However, monocular cameras suffer from a critical drawback known as scale uncertainty, leading to potential angular drift. To address this limitation, stereo camera systems have been proposed, effectively mitigating the issue of scale uncertainty. Nevertheless, their accuracy and robustness still lag behind those of LiDAR systems.

Different situations entail varying precision and system cost requirements [1]. Mapping requires high-precision sensors, and, in this study, sensitivity to cost is less critical since the equipment can be reused. In autonomous vehicle localization, as the number of vehicles increases, the processing load on the sensors also rises, often leading to errors

in feature processing and reduced SLAM accuracy. To balance precision and cost, it is necessary to use high-precision yet cost-effective sensors. An effective approach to reducing costs while maintaining accuracy is to leverage the strengths of both cameras and LiDAR. Generally, there are two methods for fusing LiDAR and cameras [2]. The first method involves synthesizing RGB images from the LiDAR map [3,4]. This approach requires solving the relative extrinsic parameters of the camera and LiDAR, but the computational load for mapping images to the point cloud is substantial, making it unsuitable for localization systems with strong real-time performance requirements. The second method involves creating landmarks in the fused map [5,6], such as by identifying lane lines to aid visual localization. However, this method lacks generality and may not perform well on certain road types, such as rural roads. Therefore, the performance of the aforementioned methods is suboptimal in complex and dynamic environments.

Calibration is required for multi-sensor fusion methods, and the accuracy of the calibration results can affect the accuracy of localization. Unlike directly fusing point clouds and images, we propose a method for localization using stereo cameras to avoid the need for sensor calibration. This method extracts geometric features from a LiDAR map previously generated by algorithms like G-LOAM [7] and then selects points with the same geometric properties in the visual image. These points satisfy both the bundle adjustment (BA) constraints and geometric constraints, as illustrated in Figure 1. Urban environments are primarily characterized by flat ground features. This work leverages our previous research on accurately identifying ground features [8] and the advantages of tensor voting [9] in spatial plane segmentation to design a tightly coupled trajectory estimation system. In summary, the contributions of this paper are outlined as follows:

(1) A 3D SLAM framework has been developed, which significantly reduces cumulative trajectory errors while enabling the construction of clear 3D reconstructions.
(2) A method that tightly couples stereo vision images and LiDAR point clouds by relying on ground information is proposed.
(3) A method for reconstructing 3D point clouds from spatial features using local bundle adjustment is employed.

The rest of this paper is structured as follows. In Section 2, we review related works concerning the evolution of SLAM and the challenges it encounters. Section 3 elucidates the proposed visual geometry and disparity calculation. The point cloud segmentation and extraction algorithm is presented in Section 4. The pose optimization is detailed in Section 5. The experimental results and future directions for this work are discussed in Section 5 and Section 6, respectively.

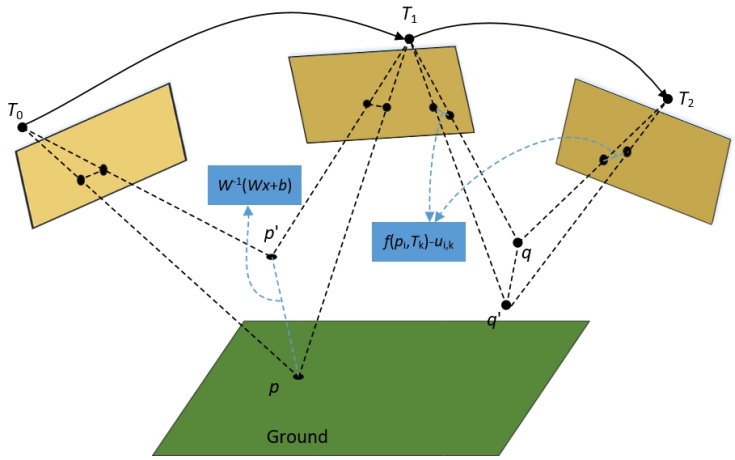


Figure 1. Pose optimization based on ground information. T, p, and q represent the transformation matrix, points on the plane, and points off the plane, respectively.

2. Related Work

In the past few decades, significant advancements have been made in both vision-based [10,11] and LiDAR-based [12] localization technologies. Methods based on cameras and LiDARs are frequently used for mobile robot localization. Vision sensors can detect the color features of the surrounding environment, while LiDARs provide precise distance information and exhibit strong robustness to varying lighting conditions.

Using laser radar cameras effectively for localization and mapping is a great choice, which can significantly enhance the applicability of algorithms. For instance, algorithms that fuse three types of sensors can effectively operate in scenes with poor lighting conditions or degraded structures. In recent years, there has been extensive literature on SLAM methods that integrate laser radar and cameras. Zhang et al. proposed DEMO [13], which utilizes depth values from laser point clouds to provide depth for some visual feature points. Compared to purely visual methods, DEMO achieves higher pose accuracy and provides higher-quality point cloud maps. Zhang et al. [14,15] further proposed the V-LOAM algorithm, which fuses multiple sensors in a unified optimization-based estimator framework to fully exploit the advantages of different sensors. V-LOAM can operate in highly dynamic scenes, demonstrating strong robustness and scene adaptability. V-LOAM is particularly prominent for its high precision and low drift in the renowned KITTI odometry evaluation dataset [16]. Graeter et al. proposed LIMO [17], which employs a novel approach to provide depth to image feature points using radar point clouds and assigns different weights to visual feature points based on semantic labels obtained from neural networks to better utilize sensor data.

Due to the relatively high cost of LiDAR-based methods and the inability of the traditional vision-based approaches to meet localization accuracy requirements, map-based visual localization has gradually emerged as a hot research topic. In [18], a monocular camera localization method utilizing a 3D prior map generated by LiDAR was proposed. The innovation lies in generating synthetic views from various poses using a GPU and subsequently computing the normalized mutual information (NMI) between these synthetic views. The pose with maximized NMI is identified as the desired pose. However, the generated map does not include height information. In a different approach, ref. [19] presented a monocular-vision-based urban localization technique focusing on road lane identification using a random forest-based edge detector. The Chamfer distance is computed between the detected edges and projected road marking points in the image space. Non-linear optimization, incorporating epipolar geometry constraints and odometry, is then employed to estimate the six-degrees-of-freedom camera pose. The authors in [20] utilized Semi-Global Block Matching (SGBM) to estimate the disparity and recover depth from stereo images. The depth from the stereo camera is matched with a prior LiDAR map, and a complete six-degrees-of-freedom camera pose is estimated by minimizing the depth residual. Some scholars have explored cloud robotic systems [21,22], with plans to apply the developed methods to future cloud robot work.

In the field of visual-LiDAR SLAM, concurrent strategies with multiple sensors, LiDAR-assisted visual SLAM, camera-supported LiDAR SLAM, and tightly coupled fusion methods have been explored [23]. LIMO SLAM [17] models local planes using LiDAR depth information and extracts depth features from camera images. Shin et al. [24] integrated visual and sparse LiDAR depth data for SLAM, employing optimized sliding window techniques and incorporating loop closure functionality, achieving significant performance across various large-scale datasets. LIC-fusion (LiDAR-Inertial-Camera) [25] ensures good localization accuracy and robustness by integrating LiDAR features, camera features, and high-frequency IMU data with online multi-sensor calibration capabilities. Chou et al. [26] proposed a tightly integrated visual-LiDAR SLAM, termed TVL-SLAM, with separate frontends for LiDAR and camera, fused in the backend for pose optimization with LiDAR loop closure. TVL-SLAM outperforms single visual and LiDAR SLAM implementations.

R2 LIVE [27] combines an error-state Kalman filter with factor-graph optimization to estimate trajectories, integrating LiDAR, IMU, and visual measurements. Its successor, R3 LIVE [28], took a further step by integrating a colored 3D map, a combination of VIO and LIO subsystems, and online photometric calibration, ensuring robustness against camera failures and LiDAR degradation scenarios, even on small onboard platforms, showcasing remarkable performance. Super Odometry [29], a brainchild of Zhao et al., emerged as an IMU-centric robust state estimation pipeline, demonstrating remarkable efficacy, particularly in challenging environments with degraded perception.

AdVLO [30] uses an attention mechanism to select key regions, fusing visual and LiDAR features for improved odometry estimation. Nowakowski [31] proposed combining YOLO and laser scanning to enhance target detection and precise distance estimation on unmanned platforms. Further, 3D cameras and LiDAR help small robots to navigate autonomously and weed in orchards, overcoming canopy navigation challenges [32]. These methods have been validated on the KITTI dataset and in some orchard scenarios using semantic or depth map fusion techniques. This paper focuses more on research aimed at improving SLAM accuracy in urban environments.

Integrating multiple sensor measurements in a sensible manner often leads to performance improvements, but algorithm designers should also be mindful of additional considerations. To fuse data from various sensors effectively, the calibration of the extrinsic parameters between different sensors is necessary, which includes the six-degrees-of-freedom (six-DoFs) relative pose transformation and time offset. Achieving accurate extrinsic parameters between sensors of different modalities typically requires complex offline calibration processes. Over prolonged usage, mechanical deformations may cause variations in the extrinsic parameters between sensors. When precise calibration of extrinsic parameters cannot be obtained, online calibration within the estimator can be conducted to mitigate the effects of poor extrinsic parameters. However, online calibration of extrinsic parameters is susceptible to degradation motion effects. For example, pure translational motion or single-axis rotational motion in visual–inertial systems can render the translation component of camera and IMU extrinsic parameters unobservable [33].

In conclusion, the landscape of visual-LiDAR SLAM is rapidly evolving, propelled by innovations in sensor fusion, optimization techniques, and robust state estimation pipelines. From the integration of LiDAR depth data with visual SLAM frameworks to the advancement of tightly coupled sensor fusion methods, researchers are pushing the boundaries of precision, robustness, and real-time performance. As the field progresses, the seamless integration of multimodal sensor data continues to unlock new possibilities, offering enhanced navigation solutions across a diverse range of environments and scenarios. The ongoing pursuit of Super Odometry and similar breakthroughs underscores the quest for ever-improving performance, particularly in challenging perceptual conditions, signifying an exciting era of exploration and discovery in visual-LiDAR SLAM research.

3. Stereo Matching and Disparity Calculation

3.1. Stereo Geometry

A parallel stereo rig is set up as described in [34,35]. As depicted in Figure 2, the world coordinates of a 3D point $P(X, Y, Z)$ are projected onto a 2D point $p(u, v)$ in the image. (u_0, v_0) represents the optical center in the projected image coordinate. The coordinates (u, v) can be expressed as follows:

$$\begin{aligned} u &= u_0 + \frac{fX - \varepsilon_i \frac{b}{2} f}{(Y+h)\sin\theta + Z\cos\theta} \\ v &= v_0 - f\tan\theta + \frac{fb}{\cos\theta[(Y+h)\cos\theta + Z\cos\theta]}. \end{aligned} \tag{1}$$

where i represents either the left or right camera. $\varepsilon_l = -1$ and $\varepsilon_r = 1$. The camera's pitch angle and height are denoted as θ and h, respectively. The stereo baseline and the measured local length are represented by b and f, respectively.

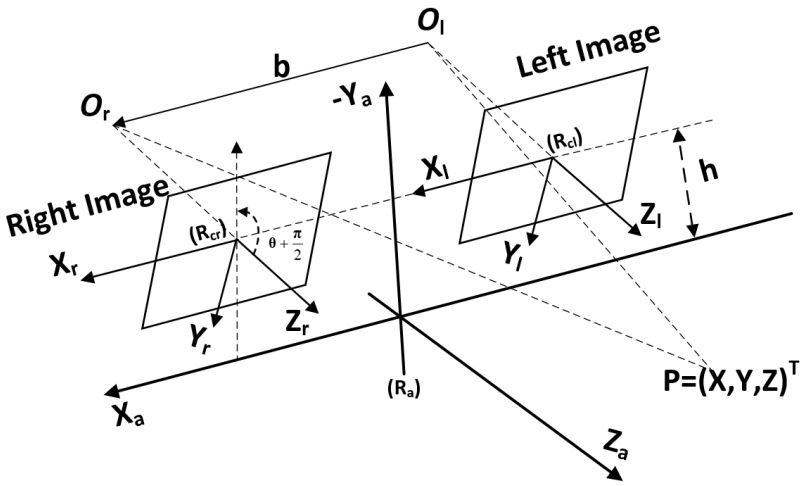


Figure 2. The stereo sensor model and the coordinate systems used [34].

Subsequently, the disparity $\Delta = u_l - u_r$ is calculated as

$$\Delta = \frac{fb}{(Y+h)\sin\theta + Z\cos\theta}. \tag{2}$$

A map that includes the disparity value for each pixel is referred to as a disparity map.

$$\Delta = \frac{b(v\cos\theta - v_0\cos\theta + f\sin\theta)}{(Y+h)}. \tag{3}$$

3.2. Positive Obstacle Removal

The fast and effective obstacle removal scheme is based on the concept of u-disparity. Peaks in the u-disparity are identified using a threshold denoted as O_b. Any point with a u-disparity value greater than O_b is classified as an obstacle point. Subsequently, a new disparity map is generated from the original one after the identified positive obstacles have been removed. The primary objective of this pre-processing step is to aid in extracting the ground profile rather than attempting to identify all obstacle pixels in the disparity map.

As depicted in Figure 3, elements such as the sky, trees, roadside features, and pedestrians are effectively eliminated through this initial positive obstacle removal process. This refinement leads to a more accurate definition of the region of interest (ROI). Consequently, it results in enhanced detection accuracy, as further discussed in the subsequent section.

3.3. Ground Detection

We use a stereo camera for plane detection, and an example of a disparity image is shown in Figure 3.

We referred to the method outlined in [34] and utilized the u-disparity and v-disparity of the disparity map for ground estimation. Specifically, the u-disparity is generated along the image rows, while the v-disparity is generated by computing the disparity histogram along the rectified image columns. Regions with peak disparitites are identified as large obstacles. To estimate the ground surface, we employed non-parametric road detection [35], a fast and efficient method suitable for non-planar ground surfaces.

In this method, the u-disparity is first calculated from the disparity map, and peak values in the u-disparity image (considered large obstacles) are removed. Points with a u-disparity greater than O_b are treated as peaks. The value of O_b is calculated using the formula $O_b = (H_o \times d)/b$ (where d represents the disparity value), with H_o being the minimum height required for an obstacle. To expedite the process, we store the calculated thresholds for each disparity as a lookup table.

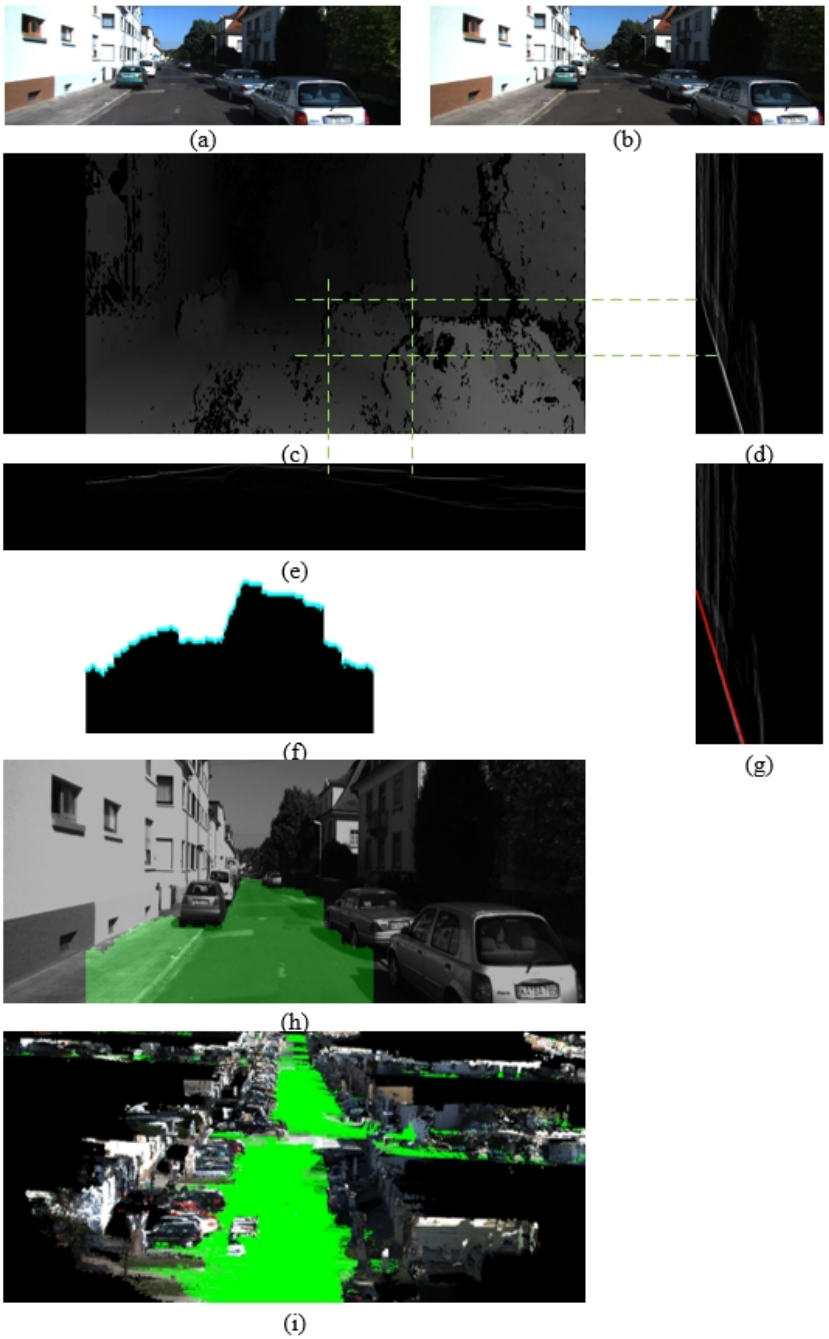


Figure 3. Region of interest extraction. (**a**) Left image. (**b**) Right image. (**c**) Disparity image. (**d**) v-disparity. (**e**) u-disparity. (**d**,**e**) are derived from (**c**). (**f**) Large obstacles removed by removing peak values from (**e**). (**g**) v-disparity based on (**f**), and red line is disparity profile of ground plane. (**h**) Detected ground plane and region of interest (RoI); RoI in red box. (**i**) City 3D reconstruction; green represents ground.

Removing large obstacles before generating the v-disparity enables more accurate ground detection. Therefore, while traversing the v-disparity row by row, we have already removed the disparity values considered as large obstacles. Through this process, we can estimate the disparity values for each row and for all pixels on the ground. The overall process is illustrated in Figure 3.

4. Groundplane Detection in LiDAR

In the pursuit of creating high-precision maps using LiDAR and GPS data, it becomes essential to extract relevant geometric information from the map. This information serves as crucial constraints for visual positioning. Through our observations, we have verified that plane features hold significant value, especially in urban environments. The extraction of these plane features from point cloud maps is a relatively straightforward process.

In our proposed method, we utilize the homogeneous coordinates of points in 2D and 3D as presented below.

$$p = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \text{and } P = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{4}$$

The camera's pose is depicted through an SE(3) transformation.

$$T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \text{where T} \in \text{SE(3)}, \tag{5}$$

which is related to its corresponding $se(3)$ Lie algebra $\xi^{\wedge}$ by exponential map

$$T(\xi) = exp(\xi^{\wedge}) \tag{6}$$

The wedge operator $\wedge$ turns $\xi \in R^6$ into a member of the $se(3)$ Lie algebra. For tracking and localization, camera pose T is iteratively updated as

$$T \leftarrow T(\xi) \cdot T \tag{7}$$

where the increment $T(\xi)$ is applied by using left multiplication convention.

The processing of the point cloud map can be broken down into two distinct steps. Initially, we employ the tensor voting algorithm, as outlined in [9,36], to compute the normal vectors for each point within the point cloud. Subsequently, in the second step, we leverage the k-means algorithm to group points with similar normals. Based on the outcomes of this clustering process, we calculate the equations defining the planes.

Tensor voting has proven to be an effective technique for determining surface normals. Its fundamental concept revolves around extracting implicit geometric characteristics from extensive and scattered point cloud data by facilitating the transfer of tensors between adjacent points. As the distance between points increases, the influence of points within the voting field gradually diminishes. Adhering to this principle, we establish the tensor kernel as follows:

$$k(d, \sigma) = e^{-\frac{d}{\sigma^2}} \tag{8}$$

The computation involves the distance metric $d = (x_i - x)^2 + (y_i - y)^2$, where (x, y) denotes the coordinates of the point receiving the vote and (x_i, y_i) represents the coordinates of the voting point. The parameter σ signifies the kernel size of the sparse voting field. Representing the input point P is achieved through a second-order symmetric semi-positive definite tensor, denoted as S. This tensor can be expressed as a positive semidefinite matrix $S_{3\times3}$ due to the equivalence between tensors and matrices. Furthermore, it can be decomposed into three distinct components, taking the following forms:

$$\begin{aligned} S &= \lambda_1 \hat{e}_1 \hat{e}_1^T + \lambda_2 \hat{e}_2 \hat{e}_2^T + \lambda_3 \hat{e}_3 \hat{e}_3^T \\ &= (\lambda_1 - \lambda_2)\hat{e}_1 \hat{e}_1^T + \qquad (stick\ component) \\ &\quad (\lambda_2 - \lambda_3)(\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T) + \quad (plate\ component) \\ &\quad \lambda_3(\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T + \hat{e}_3 \hat{e}_3^T) \qquad (ball\ component) \end{aligned} \tag{9}$$

Following the tensor voting framework, we can obtain the normal vectors for each point. According to Equation (2), we can identify the points on the ground that conform to the following characteristics:

$$P_t = (\lambda_2 - \lambda_3)(\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T) \tag{10}$$

5. Pose Optimization

Visual points can be classified into two categories: points p lying on the ground and points q not on the ground. In the LiDAR coordinate system, the distance from a planar point p to a spatial point p' is $|n|_2^{-1}(n^T p + b)$. In the visual localization framework, point q only needs to minimize the reprojection error $(f(q_i, T_k) - u_{i,k})$ (where f is the mapping function from 3D to 2D), while point P must minimize both the reprojection error and the coplanarity error.

Traditional visual localization consists of three main stages. Firstly, the pose between two frames is calculated by matching features between them. This stage involves feature extraction and matching, typically using local invariant features such as ORB, SIFT, or SURF to describe key points in the images. By comparing descriptors, correspondences between frames are found, enabling the computation of their relative pose. Subsequently, depth information is reconstructed through triangulation. This step utilizes the camera projection relationship from two viewpoints. With known camera parameters and matched feature points, depth values are computed to build a 3D point cloud of the scene. Finally, bundle adjustment is performed based on the map points and corresponding frames. Bundle adjustment optimizes camera poses and 3D point cloud positions to minimize reprojection error, enhancing the accuracy and consistency of the entire system.

In this paper, our approach introduces projection constraints for ground points in addition to bundle adjustment during optimization. Traditional bundle adjustment methods consider only the reprojection error of feature points, neglecting the coplanarity constraints of ground points, which can lead to decreased accuracy and robustness of localization in dynamic environments or scenes rich in planar ground information. To address this issue, we incorporate additional projection constraints for ground points during the optimization process, leveraging ground plane information to enhance localization accuracy.

We employ an incremental sliding window method to connect observed points with the current point in each incoming frame. In this process, the target points within the frame are optimized concurrently with the observed images to determine a maximum six-degrees-of-freedom (six-DOFs) pose. This approach enables continuous and precise tracking of the camera's movement, enhancing the overall robustness of the localization process.

We represent the Lie algebra corresponding to the rotation matrix R and translation vector t as $\xi^\wedge$. The Lie algebra provides a mathematical framework for representing the infinitesimal rotations and translations, enabling efficient optimization in the parameter space. This compact representation simplifies the process of updating the camera poses during the optimization. Here, p represents the observed map point, which is a 3D feature point identified and tracked across multiple image frames. These map points are critical for determining the geometric structure of the scene. The provided formula represents the error resulting from the observation of the k-th point in the j-th frame. This error is the difference between the actual observed position of the point in the image and its reprojected position based on the current estimates of the camera pose and the 3D point position.

The cost function, which we aim to minimize during bundle adjustment, is expressed as follows:

$$E_{ba} = ||(u_i - Kexp(\xi^\wedge)p_k)^T Q_{k,j}^{-1}(u_i - Kexp(\xi^\wedge)p_k)||_2. \quad (11)$$

where u_i represents the observed 2D pixel coordinates of point i. K is the camera intrinsic matrix, used to map 3D points from the camera coordinate system to the 2D image plane. Denotes the pose transformation representing camera motion, expressed using the exponential map from Lie algebra, where ξ represents the motion parameters.

For the points located on the plane, we introduce plane constraints as additional constraints using the sliding window algorithm. In the first phase, BA is applied to these points. As more plane information is introduced, the number of plane points gradually increases, and corresponding constraints are introduced, leading to more accurate pose estimation. The ground equation can be represented as $Wx + b = 0$, and its projection cost function is defined as follows:

$$E_p = ||(\frac{1}{|n|_2}(n^T T_i p_j + b))^T R_j^{-1}(\frac{1}{|n|_2}(n^T T_i p_j + b))||_2^2. \quad (12)$$

where n represents a weight used to describe the normal vector of the plane and b is a bias vector used to adjust the position of the plane. They are constrained by the vehicle pose T_i.

We adopted a commonly used graph optimization method to address the optimization problem in visual localization, which is based on keyframes. We constructed a graph optimization model centered on the optimization problem, as shown in Figure 4, where each edge in the graph represents different constraints. The factor graph formulation is highly intuitive and has the advantage of enabling efficient implementations of batch and incremental solvers. Our optimization system includes both BA constraints and plane constraints.

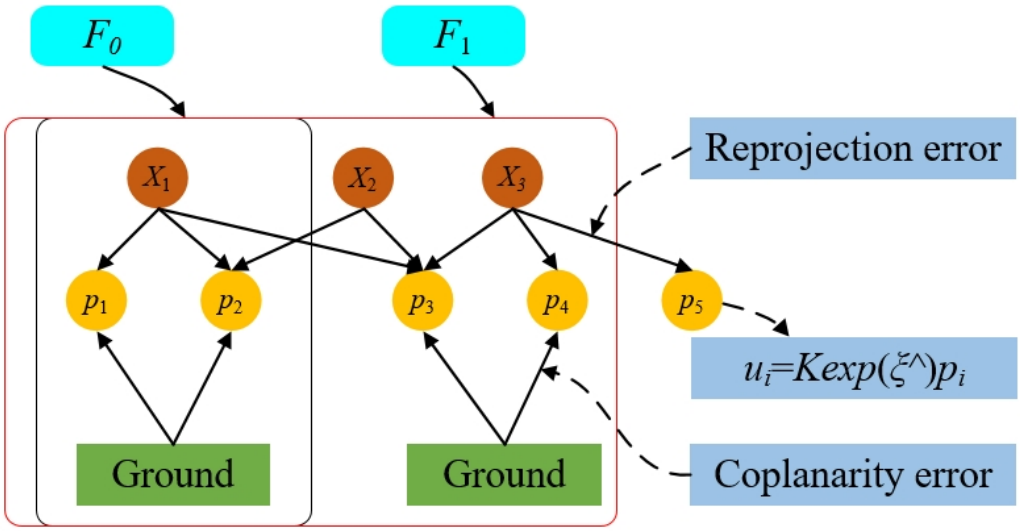


Figure 4. Graph-structure optimization. P represents the nodes of visual points, and X represents the pose of the frame. "Ground" denotes the ground information extracted from the 3D reconstruction.

In our model, we denote p as the node representing visual points, X as the node representing the frame's pose, and p_g as the node representing points on the plane extracted from the ground map. We also define the error function $r_k(p_k, T)$ that quantifies the discrepancy between p_k and the observed point, and we employ an edge to represent this relationship. Similarly, $r_g(p_k, p_g)$ is defined as the cost function that projects a point onto the plane, analogous to $r_k(p_k, T)$, and we also represent this relationship using an edge. We can utilize Equation (13) to formulate the optimization problem:

$$E(p) = \sum_{k=1}^{K} \sum_{j \in E(k)} r_g(p_k, p_g) Q_{k,j}^{-1} r_l(p_k, p_g) + \sum_{p \in \pi_i} r_g(p_k, p_g)^T R_j^{-1} r_l(p_k, p_g) \tag{13}$$

The Levenberg–Marquardt scheme is used to optimize the function, and the residual around the current state is linearized.

$$r(p \oplus \delta s) = r(s) + J_s \delta s. \tag{14}$$

In the function

$$J_p = \frac{dr(p \oplus \delta p)}{2}|_{\delta p = 0}. \tag{15}$$

The error function $E(s)$ is approximated with a quadratic function for the current state s:

$$\begin{aligned} E(p \oplus \delta p) &= E_p + \delta p^T b_p + \frac{1}{2} \delta p^T H_s \delta b_p \\ b_p &= J_p^T W r(p) \\ H_p &= J_p^T W J(p) \end{aligned} \tag{16}$$

where b_p is the Jcaobian matrix and H_p is the Hessian matrix. This function updated the state $p \leftarrow p \oplus \delta p$ by minimizing $\delta p = -H^{-1} b_p$. We repeat this update and linearization process until convergence.

6. Experimental Results

This section provides a summary of the results obtained using our proposed method. Our goal is to conduct a comparative analysis of the current state-of-the-art systems, with a particular focus on accuracy. This article uses the left disparity image as the ground for image detection while mapping the LiDAR point cloud onto the left image. We conducted experiments to evaluate the performance of our proposed approach using the training dataset from the publicly available KITTI [37] benchmark. The dataset comprises sequences 00–10, each accompanied by ground truth trajectories. Our analysis focused on assessing the accuracy of our method. The experiments were conducted on a computer equipped with an Intel Core i5-12400F @ 4.2GHz processor and 32GB of memory, running a 64-bit Linux operating system.

6.1. Comparison with Others

This paper compares the ORB-SLAM3 [38] stereo module, F-LOAM [39], LOAM [40], and LeGO-LOAM [41], as shown in Table 1. For fairness, we referenced [42] to provide the trajectory estimation of F-LOAM, LOAM, and LeGO-LOAM on sequences 00–10 of the KITTI dataset. Compared to these methods, our approach has achieved an average improvement of at least 63.64% in RMSE and 65.51% in standard deviation when evaluated using the ATE method. When evaluated using the RPE method, our approach has achieved an average improvement of at least 52.15% in RMSE and 59.08% in standard deviation. Figure 5 illustrates the trajectories of 11 representative sequences. The gray dashed line in the figure represents the ground truth of the dataset, while the colored solid line indicates the position estimated by our method. The color transition from red to blue illustrates the error in our method's position estimation. The error range is depicted on the right side of each subplot. Figure 6 displays the trajectory error of our algorithm across these 11 representative sequences. The assessment system initially aligns actual and estimated values by utilizing the timestamps of the poses. Subsequently, it computes the disparity between each pair of poses. The figure depicts the absolute trajectory error, mean error,

median error, and RMSE in meters. It can be seen that our algorithm has better effects than ORB-SLAM3 in multiple sequences and demonstrates better robustness.

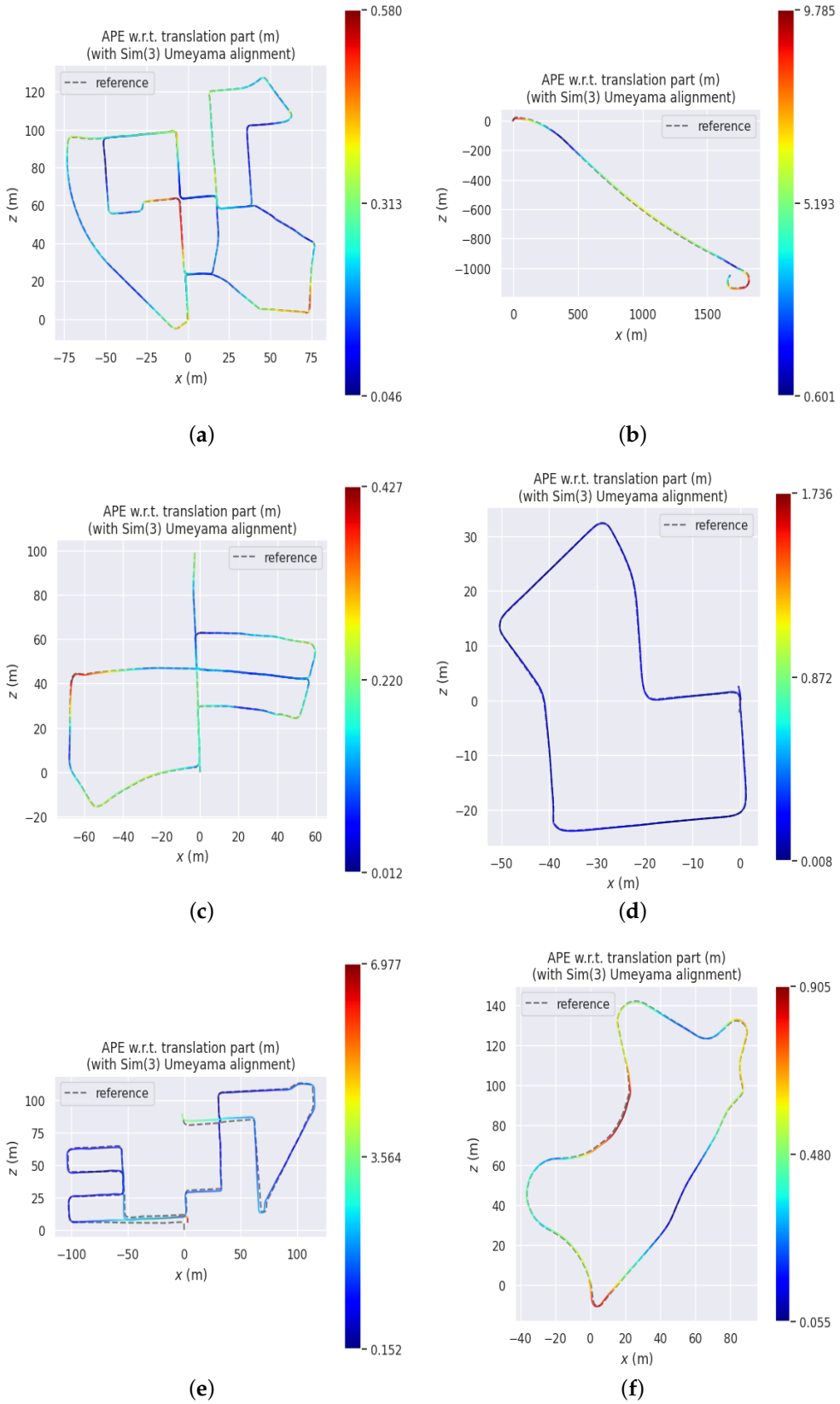


Figure 5. Trajectory estimates in the KITTI dataset. (**a**) 00. (**b**) 01. (**c**) 05. (**d**) 07. (**e**) 08. (**f**) 09.

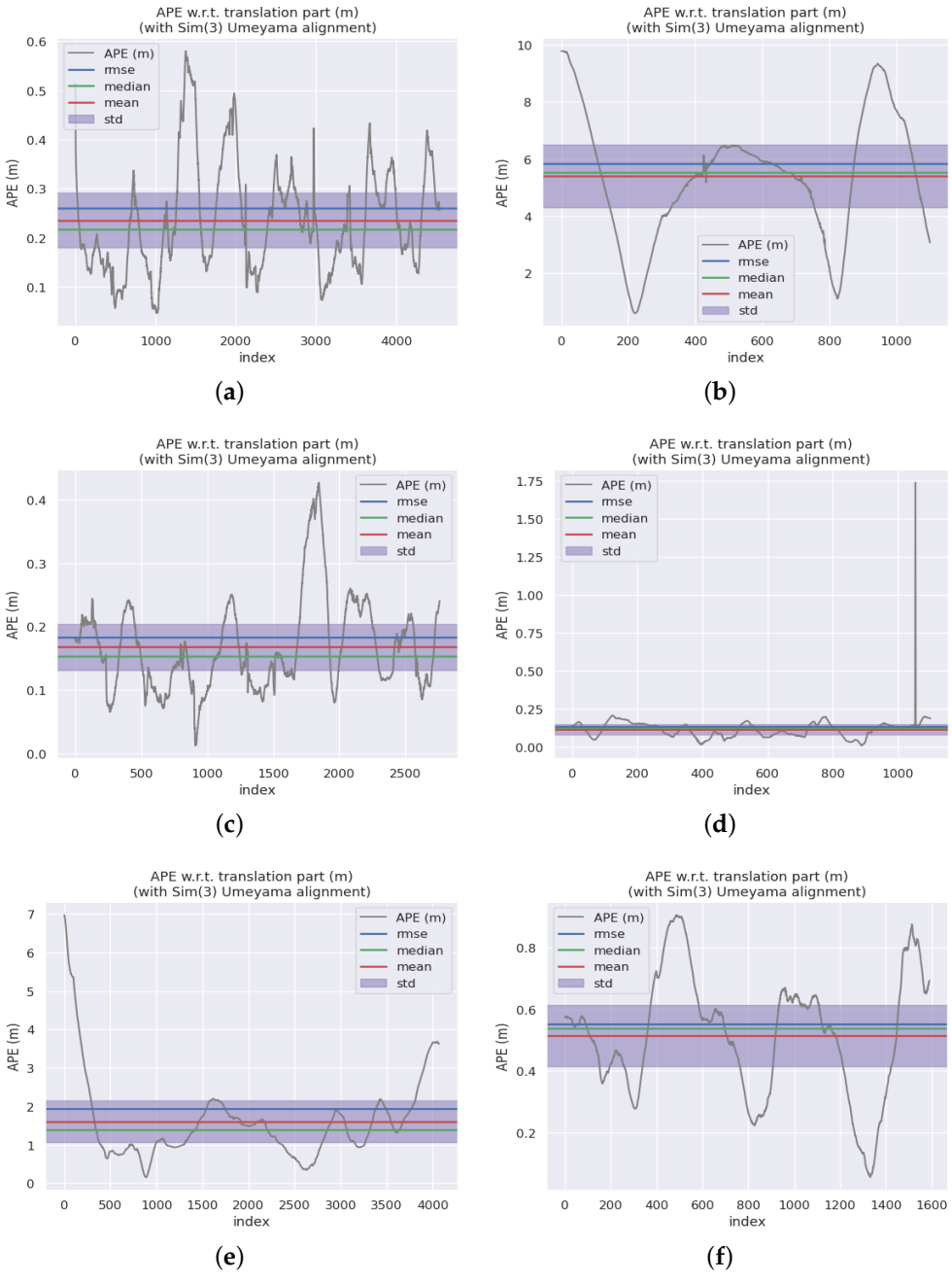


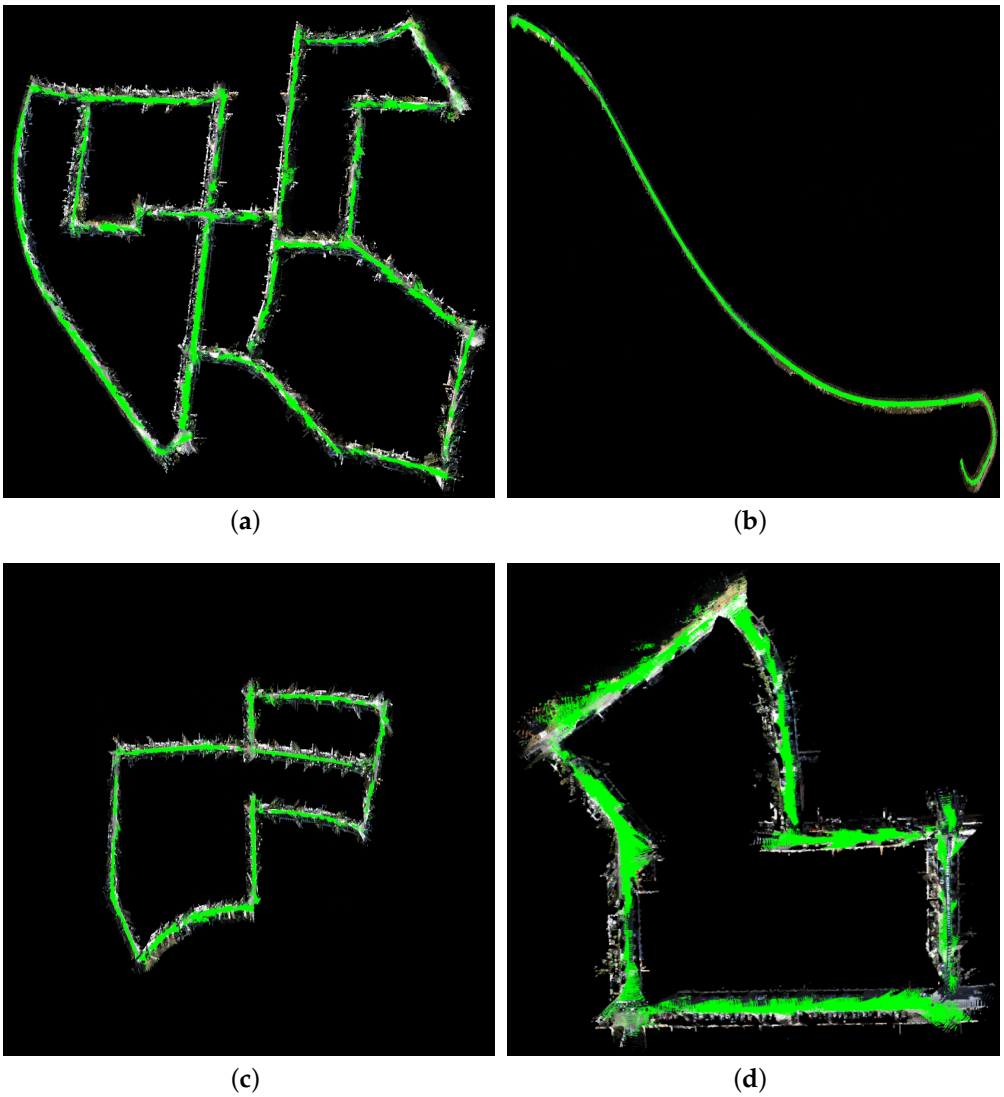
Figure 6. High-definition display of point clouds for some streets in the 00 sequence. (**a**) 00. (**b**) 01. (**c**) 05. (**d**) 07. (**e**) 08. (**f**) 09.

Table 1. Results of metrics' absolute trajectory error (ATE) [M].

Sequence	ORB-SLAM3	Fast-LOAM	LeGO-LOAM	LOAM	Ours
00	**0.22**	0.96	5.03	0.96	0.26
01	2.09	2.80	21.02	2.68	**1.52**
02	10.82	1.57	2.80	5.04	**0.59**
03	0.43	1.09	1.32	1.16	**0.09**
04	0.18	1.43	1.63	1.39	**0.04**
05	0.38	0.80	0.93	0.71	**0.18**
06	0.36	0.72	0.84	0.72	**0.23**
07	0.39	0.55	0.77	0.54	**0.13**
08	2.87	**1.16**	1.58	1.18	1.93
09	0.88	1.29	1.48	1.21	**0.55**
10	1.27	1.77	1.86	1.61	**0.94**

6.2. 3D Reconstruction Experiments

We conducted a 3D reconstruction experiment, as shown in Figure 7, which displays the overall point cloud of the corresponding dataset. The detected ground information is marked in green. According to Figure 5, the scene has a maximum size of 2000 m by 1200 m. As shown in Figure 8, we present a high-definition point cloud of a section of the street from sequence 00, where the clarity of the point cloud depends on the accuracy of the estimated trajectory. Thanks to the accuracy of the road surface extraction method [8] and the contribution of tensor voting [9] in the segmentation of spatial point clouds, the variance optimization constructed in our system has led to higher trajectory accuracy. It can be observed that we maintained significant clarity in a large-scale 3D reconstruction scene, demonstrating the high accuracy of our method.



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

(**c**) (**d**)

Figure 7. *Cont.*



(**e**) (**f**)

Figure 7. 3D reconstruction based on road constraints, where green represents the road. (**a**) 00. (**b**) 01. (**c**) 05. (**d**) 07. (**e**) 08. (**f**) 09.

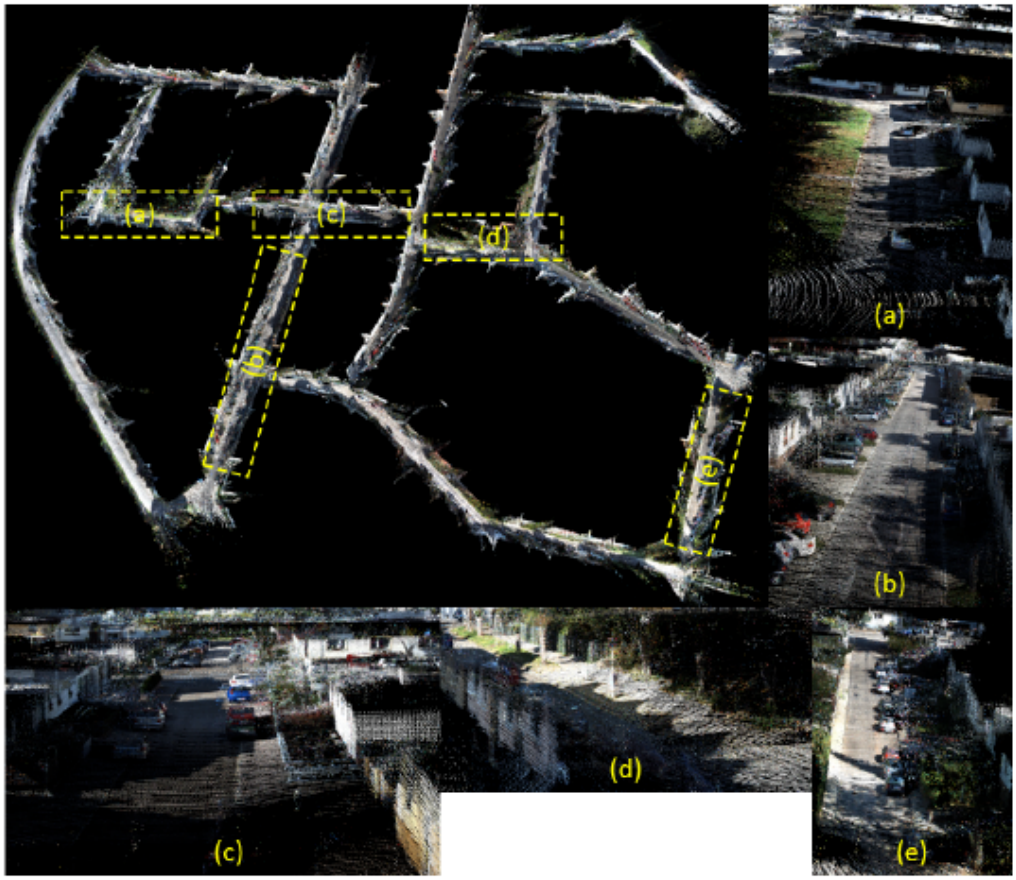


Figure 8. High-definition display of point clouds for some streets in the 00 sequence.The image in the top left corner is a 3D reconstruction of the entire city, and the other images depict details of its streets (**a**–**e**).

7. Conclusions and Future Work

This paper proposes a trajectory estimation and mapping method based on ground information, with a focus on urban and flat road environments. The method integrates

stereo vision and LiDAR data. Compared to the standalone LiDAR or vision methods, we achieve higher trajectory estimation accuracy. The method demonstrates stable performance in complex environments, achieving decimeter-level localization accuracy even at the urban scale while also enabling clear 3D reconstruction.

To leverage the advantages of stereo cameras and LiDAR while minimizing the computational costs, we align the ground information between the LiDAR map and visual images. Geometric features are extracted from the point cloud map and images through tensor voting to establish coplanarity constraints. Pose optimization is achieved through graph optimization and local window optimization.

In our experimental evaluation, we assessed the performance of our method in the urban environment of the KITTI dataset. The results indicate that our method outperforms the current state-of-the-art systems in both robustness and accuracy, showcasing a significant advantage in this particular setting. Future work will focus on optimizing the inference speed, exploring better integration of RGB and point-wise features, and incorporating additional operations on point clouds to further enhance our detection framework. Additionally, we will conduct further research on off-road scenarios, such as open areas with few features and bumpy terrains.

Author Contributions: Conceptualization, T.S. and L.C.; methodology, L.C.; software and validation, L.C.; formal analysis, X.Y.; investigation, X.Y.; writing—original draft preparation, T.S.; writing—review and editing, T.S.; supervision, Y.L. and Y.Z.; project administration, Y.L.; funding acquisition, T.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by Digital Workshop Industrial Control Network Security Project (No. S11180B42314) and the Research on Higher Education Teaching Reform of Nanjing Normal University in 2023 (No.18122000462323).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are openly available in TUM at https://www.cvlibs.net/datasets/kitti (accessed on 5 June 2024).

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

References

1. Deutsch, I.; Liu, M.; Siegwart, R. A framework for multi-robot pose graph SLAM. In Proceedings of the 2016 IEEE International Conference on Real-Time Computing and Robotics (RCAR), Angkor Wat, Cambodia, 6–10 June 2016; pp. 567–572.
2. Liu, M.; Wang, L.; Siegwart, R. DP-Fusion: A generic framework for online multi sensor recognition. In Proceedings of the 2012 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Hamburg, Germany, 13–15 September 2012; pp. 7–12.
3. Steder, B.; Ruhnke, M.; Grzonka, S.; Burgard, W. Place recognition in 3D scans using a combination of bag of words and point feature based relative pose estimation. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, 25–30 September 2011; pp. 1249–1255.
4. Pascoe, G.; Maddern, W.; Newman, P. Direct visual localisation and calibration for road vehicles in changing city environments. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Santiago, Chile, 7–13 December 2015; pp. 9–16.
5. Wang, R. 3D building modeling using images and LiDAR: A review. *Int. J. Image Data Fusion* **2013**, *4*, 273–292. [CrossRef]
6. Ding, X.; Wang, Y.; Li, D.; Tang, L.; Yin, H.; Xiong, R. Laser map aided visual inertial localization in changing environment. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 4794–4801.
7. Zheng, L.; Zhu, Y.; Xue, B.; Liu, M.; Fan, R. Low-cost gps-aided lidar state estimation and map building. In Proceedings of the 2019 IEEE International Conference on Imaging Systems and Techniques (IST), Abu Dhabi, United Arab Emirates, 9–10 December 2019; pp. 1–6.
8. Sun, T.; Pan, W.; Wang, Y.; Liu, Y. Region of Interest Constrained Negative Obstacle Detection and Tracking with a Stereo Camera. *IEEE Sens. J.* **2022**, *22*, 3616–3625. [CrossRef]
9. Liu, M. Efficient segmentation and plane modeling of point-cloud for structured environment by normal clustering and tensor voting. In Proceedings of the 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014), Bali, Indonesia, 5–10 December 2014; pp. 1805–1810.

10. Liu, M.; Pradalier, C.; Chen, Q.; Siegwart, R. A bearing-only 2D/3D-homing method under a visual servoing framework. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, Alaska, 3–8 May 2010; pp. 4062–4067.
11. Liu, M.; Pradalier, C.; Pomerleau, F.; Siegwart, R. Scale-only visual homing from an omnidirectional camera. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation, St Paul, MN, USA, 14–18 May 2012; pp. 3944–3949.
12. Bhutta, M.U.M.; Liu, M. PCR-Pro: 3D sparse and different scale point clouds registration and robust estimation of information matrix for pose graph SLAM. In Proceedings of the 2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Tianjin, China, 19–23 July 2018; pp. 354–359.
13. Zhang, J.; Kaess, M.; Singh, S. Real-time depth enhanced monocular odometry. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 4973–4980.
14. Zhang, J.; Singh, S. Visual-lidar odometry and mapping: Low-drift, robust, and fast. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 2174–2181.
15. Zhang, J.; Singh, S. Laser–visual–inertial odometry and mapping with high robustness and low drift. *J. Field Robot.* **2018**, *35*, 1242–1264. [CrossRef]
16. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets robotics: The kitti dataset. *Int. J. Robot. Res.* **2013**, *32*, 1231–1237. [CrossRef]
17. Graeter, J.; Wilczynski, A.; Lauer, M. Limo: Lidar-monocular visual odometry. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 7872–7879.
18. Wolcott, R.W.; Eustice, R.M. Visual localization within lidar maps for automated urban driving. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 176–183.
19. Lu, Y.; Huang, J.; Chen, Y.T.; Heisele, B. Monocular localization in urban environments using road markings. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017; pp. 468–474.
20. Kim, Y.; Jeong, J.; Kim, A. Stereo camera localization in 3d lidar maps. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1–9.
21. Wang, L.; Liu, M.; Meng, M.Q.H. A hierarchical auction-based mechanism for real-time resource allocation in cloud robotic systems. *IEEE Trans. Cybern.* **2016**, *47*, 473–484. [CrossRef] [PubMed]
22. Liu, B.; Wang, L.; Liu, M. Lifelong federated reinforcement learning: A learning architecture for navigation in cloud robotic systems. *IEEE Robot. Autom. Lett.* **2019**, *4*, 4555–4562. [CrossRef]
23. Debeunne, C.; Vivet, D. A review of visual-LiDAR fusion based simultaneous localization and mapping. *Sensors* **2020**, *20*, 2068. [CrossRef] [PubMed]
24. Shin, Y.S.; Park, Y.S.; Kim, A. Direct visual slam using sparse depth for camera-lidar system. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 5144–5151.
25. Zuo, X.; Geneva, P.; Lee, W.; Liu, Y.; Huang, G. Lic-fusion: Lidar-inertial-camera odometry. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), The Venetian Macao, Macau, 3–8 November 2019; pp. 5848–5854.
26. Chou, C.C.; Chou, C.F. Efficient and accurate tightly-coupled visual-lidar slam. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 14509–14523. [CrossRef]
27. Lin, J.; Zheng, C.; Xu, W.; Zhang, F. R2 LIVE: A Robust, Real-Time, LiDAR-Inertial-Visual Tightly-Coupled State Estimator and Mapping. *IEEE Robot. Autom. Lett.* **2021**, *6*, 7469–7476. [CrossRef]
28. Lin, J.; Zhang, F. R3 LIVE++: A Robust, Real-time, Radiance reconstruction package with a tightly-coupled LiDAR-Inertial-Visual state Estimator. *arXiv* **2022**, arXiv:2209.03666.
29. Zhao, S.; Zhang, H.; Wang, P.; Nogueira, L.; Scherer, S. Super odometry: IMU-centric LiDAR-visual-inertial estimator for challenging environments. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 8729–8736.
30. Yin, H.; Deng, L.; Chen, Z.; Chen, B.; Sun, T.; Yusen, X.; Xiao, J.; Fu, Y.; Deng, S.; Li, X. LSMD-Net: LiDAR-Stereo Fusion with Mixture Density Network for Depth Sensing. In Proceedings of the Asian Conference on Computer Vision, Macao, China, 4–8 December 2022; pp. 552–568.
31. Nowakowski, M.; Kurylo, J.; Braun, J.; Berger, G.S.; Mendes, J.; Lima, J. Using LiDAR Data as Image for AI to Recognize Objects in the Mobile Robot Operational Environment. In Proceedings of the International Conference on Optimization, Learning Algorithms and Applications, Ponta Delgada, Portugal, 27–29 September 2023; Springer: Berlin/Heidelberg, Germany, 2023; pp. 118–131.
32. Sampurno, R.M.; Ahamed, T. Challenges in Orchard Weed Management: Perspectives on the Use of 3D Cameras and LiDAR to Develop a Low-Cost Small-Scale Robotic Weeder. In *IoT and AI in Agriculture: Smart Automation Systems for increasing Agricultural Productivity to Achieve SDGs and Society 5.0*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 229–247.
33. Yang, Y.; Geneva, P.; Eckenhoff, K.; Huang, G. Degenerate motion analysis for aided ins with online spatial and temporal sensor calibration. *IEEE Robot. Autom. Lett.* **2019**, *4*, 2070–2077. [CrossRef]
34. Labayrade, R.; Aubert, D.; Tarel, J.P. Real time obstacle detection in stereovision on non flat road geometry through "v-disparity" representation. In Proceedings of the Intelligent Vehicle Symposium, Versailles, France, 17–21 June 2002; Volume 2, pp. 646–651.

35. Wu, M.; Zhou, C.; Srikanthan, T. Robust and low complexity obstacle detection and tracking. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 1249–1254.
36. Liu, M.; Pomerleau, F.; Colas, F.; Siegwart, R. Normal estimation for pointcloud using GPU based sparse tensor voting. In Proceedings of the 2012 IEEE International Conference on Robotics and Biomimetics (ROBIO), Guangzhou, China, 11–14 December 2012; pp. 91–96.
37. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The kitti vision benchmark suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.
38. Campos, C.; Elvira, R.; Rodríguez, J.J.G.; Montiel, J.M.; Tardós, J.D. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Trans. Robot.* **2021**, *37*, 1874–1890. [CrossRef]
39. Wang, H.; Wang, C.; Chen, C.L.; Xie, L. F-loam: Fast lidar odometry and mapping. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 4390–4396.
40. Zhang, J.; Singh, S. LOAM: Lidar odometry and mapping in real-time. In Proceedings of the Robotics: Science and Systems, Berkeley, CA, USA, 12–16 July 2014; Volume 2, pp. 1–9.
41. Shan, T.; Englot, B. Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 4758–4765.
42. Yuan, Z.; Wang, Q.; Cheng, K.; Hao, T.; Yang, X. SDV-LOAM: Semi-direct visual–LiDAR Odometry and mapping. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 11203–11220. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.