IEEE Robotics and Automation Letters (RAL) paper presented at
2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

RSS: Robust Stereo SLAM with Novel Extraction and Full Exploitation of Plane Features

Haolin Wang[1,2], Hao Wei[1*], Zewen Xu[1,2], Zeren Lv[3], Pengju Zhang[1], Ning An[4], Fulin Tang[1] and Yihong Wu[1,2*]

***Abstract*—Planar structures, prevalent in man-made environments, can be observed by a camera for significant periods of time due to their large spatial presence. These structures provide strong planar regularities for Simultaneous Localization and Mapping (SLAM) systems, facilitating long-term navigation. Therefore, we propose a novel point-plane-based stereo SLAM system, fully regularized by plane features within a unified nonlinear optimization framework. The core of our method is an accurate and efficient stereo plane extraction algorithm with strict 2D and 3D outlier rejection mechanisms, effectively extracting main planes from robust stereo correspondences and enabling real-time point-plane association. Furthermore, we introduce a novel optimization formulation, incorporating geometric feature (point and plane) and across-feature (point-on-plane) constraints that promote each other through the mutual constraints between associated point and plane features, which fully exploits plane constraints to improve the performance of SLAM system. The proposed plane extraction algorithm is evaluated on the EuRoC MAV dataset, achieving significant improvements in number, accuracy, reliability, and efficiency over the state-of-the-art (SOTA) stereo point-plane-based system [1]. The results of an ablation study on two public datasets show that the proposed SLAM system outperforms [1] in both accuracy and robustness, and further demonstrate the mutual enhancement between the two types of constraints.**

***Index Terms*—SLAM, Localization.**

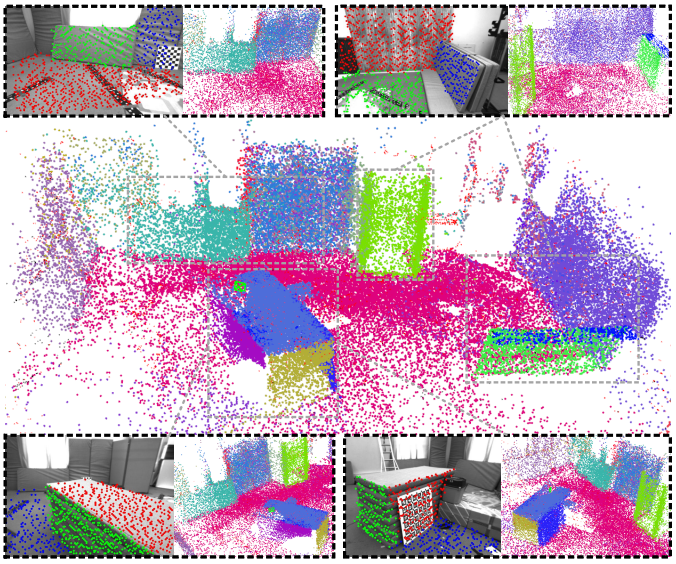


Fig. 1. Results of the proposed SLAM system on the Vicon Room 1 01 sequence of the EuRoC MAV dataset, where 3D map points belonging to the same plane have identical colors, whereas different planes are represented by distinct colors. The 2D support points used for plane extraction are also shown on the upper and lower sides.

I. Introduction

VISUAL SLAM has emerged as a foundational technology for various applications such as robot, autonomous cars, augmented reality and virtual reality. The majority of existing visual SLAM systems rely on point features to estimate camera poses, and are unable to use high-level image features as well as structural regularities information between features, which are prevalent in man-made environments [2]. To take advantage of the above regularities, some researchers focus on line and plane features [3]–[6]. Compared with line features, planes have a larger spatial presence, which is beneficial for long-term navigation and map representation. Therefore, combining point and plane features to improve the accuracy and robustness of SLAM systems has been investigated in many studies [7]–[9]. For plane-based systems, the quality of plane extraction and the approach to optimization formulation are critically important to their performance.

Plane extraction is the most critical step for plane-based SLAM, whose effect influences the performance of SLAM systems. It is easy to extract planes from point cloud [10], which is commonly acquired from an RGB-D camera [8], [11] or a LiDAR sensor [7], [9]. In contrast, plane extraction with a monocular camera is difficult because of the limited 3D information [5]. Some researchers extracted planes from sparse point landmarks [2], [3], [12], which requires accumulating a certain number of point features before extracting planes, leading to the absence of plane information in the initial frames. Unlike monocular cameras, stereo cameras can obtain dense depth images directly through stereo matching [13], enabling plane extraction via RGB-D-based methods,

Manuscript received: December 19, 2023; Revised February 4, 2024; Accepted March 27, 2024.

This paper was recommended for publication by Editor Javier Civera upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the National Natural Science Foundation of China under Grant No. 62202468 and a SINOPEC Research Project. *(Corresponding authors: Hao Wei and Yihong Wu.)*

Haolin Wang, Hao Wei, Zewen Xu, Pengju Zhang, Fulin Tang and Yihong Wu are with the State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China. Haolin Wang, Zewen Xu and Yihong Wu are also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100190, China (e-mail: {wanghaolin2023; weihao2019; xuzewen2020; pengju.zhang; fulin.tang; yihong.wu}@ia.ac.cn).

Zeren Lv is with the College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China (e-mail: 2022210463@buct.edu.cn).

Ning An is with the Research Institute of Mine Big Data, China Coal Research Institute, Beijing 100013, China (e-mail: ning.an.010@foxmail.com).

Digital Object Identifier (DOI): see top of this page.

but this process is very time-consuming, and prone to low quality in challenging scenes. Therefore, Rosinol et al. [14], [15] proposed a stereo VIO system that extracts planes from incrementally constructed 3D meshes, while enforcing point-on-plane structural regularities, yet meshes constructed from sparse point landmarks contain less structural information, compromising the number, accuracy, and reliability of the planes. Zhang et al. [1] extracted planes from stereo images based on intersecting lines that meet specific geometric constraints: angle thresholds, distance criteria, and coplanarity of endpoints. However, intersecting lines are easily affected by noise, and this method is prone to fail in scenes lacking clear edges or with cluttered lines. Others used a neural network to extract planes [16], [17]. Li et al. [16] employed a two-stage plane detection strategy using a neural network for plane segmentation followed by RANSAC outlier removal with stereo cameras. However, deep learning methods typically require extensive training data as well as additional computational resources, and their generalizability remains to be established. To enhance the accuracy and efficiency of plane extraction, we employ new Sobel-based support points for their rich, uniform, and robustly matched characteristics, offering more abundant structural information than the commonly used sparse point landmarks and intersecting lines, as shown in Fig. 1. Compared with methods that merely detect co-planar relationships among landmarks [2], [3], [12], [14], [15] or that fail to perform point-plane associations [1], we propose a novel plane extraction pipeline, which integrates comprehensive region-growing strategies with rigorous outlier rejection mechanisms to fully leverage structural information. This pipeline accurately and efficiently clusters coplanar triangles within a 3D mesh, followed by RANSAC outlier removal, achieving real-time, single-frame, multi-plane extraction without the need for costly neural networks, dense depth maps, or GPU acceleration, while also ensuring efficient and accurate point-plane association.

Planes as long-lived features with extensive spatial presence in man-made environments, are highly favorable for long-term robot navigation [2], imposing various constraints on SLAM systems, such as homography, plane reprojection, and point-on-plane constraints. Therefore, many researchers leverage plane features to enhance SLAM systems. Any two images corresponding to the same 3D planes are related by a homography. In [18]–[20], homography constraints are applied to improve the accuracy and reduce the complexity in pose estimation, yet these approaches cannot constrain the scale, leading to accumulated scale errors in continuous pose estimation, thereby causing significant scale drift. Besides, planes can also be used as geometry features [1], [4], [21]. Dai et al. [4] designed a similarity cost metric to determine the correspondences of two plane sets and derived the reprojection model for plane features. Additionally, planar regularities can be enforced to improve pose estimation accuracy through point-on-plane constraints, which can be achieved by computing the distance error between points and their associated plane [2], [3], or enforcing that in-plane points lie exactly on their associated plane [12], [16], [22], but the latter approaches, in some challenging scenes, may exhibit negligible improvement and even degradation, due to their increased susceptibility to noise and incorrect point-plane association. Existing point-plane-based stereo SLAM systems either lack the necessary geometric information for point-plane association, being confined to merely using plane reprojection constraints [1], or lack an accurate and efficient plane extraction algorithm, being confined to merely using point-on-plane constraints [15], [16]. To fully exploit the plane constraints, we propose a new optimization formulation that simultaneously integrates reprojection errors and point-on-plane errors into a unified non-linear optimization framework. This allows the associated point and plane to mutually constrain each other through across-feature constraints (point-on-plane), thereby resulting in more accurate point and plane landmarks being used to construct geometric feature constraints (point and plane). Two types of constraints continuously promote each other, thereby fully exploiting plane constraints to improve the accuracy and robustness of the SLAM system.

The main contributions are summarized as follows:

1) A novel and effective stereo plane extraction algorithm is proposed, which exploits uniform and abundant Sobel-based support points rather than costly stereo matching results or sparse point landmarks to ensure both efficiency and accuracy. In addition, strict 2D and 3D outlier rejection mechanisms are utilized to calculate plane parameters accurately. We make the code publicly available for the community at https://github.com/casia-rvg/RSS_PE.
2) An accurate and robust point-plane-based stereo SLAM system is developed, where both the geometric feature (point and plane) and across-feature (point-on-plane) constraints are exploited to improve pose estimation. Furthermore, experimental results demonstrate the two types of constraints can promote each other.
3) Experiments on two public datasets show that the proposed SLAM system outperforms the SOTA stereo point-plane-based system in both accuracy and robustness. Besides, the stereo plane extraction achieves significant improvements in terms of number, accuracy, reliability, and efficiency compared with SOTA method.

II. Robust Stereo Correspondences based Plane Extraction

To accurately and efficiently extract planes, we introduce a novel plane extraction method based on robust stereo correspondences. The proposed method consists of three components: support point extraction, 3D mesh generation, and plane extraction, as illustrated in Fig. 2.

A. Support Point Extraction

Sobel Filtering: Given a stereo pair, we first perform distortion correction. Then, Sobel filtering is applied to both left and right images using 3×3 Sobel masks to obtain the horizontal and vertical Sobel filter gradients G_x and G_y for each pixel. Finally, we select 12 horizontal gradients (taking samples twice at the central position) and 4 vertical gradients from the surrounding 5×5 neighborhood of

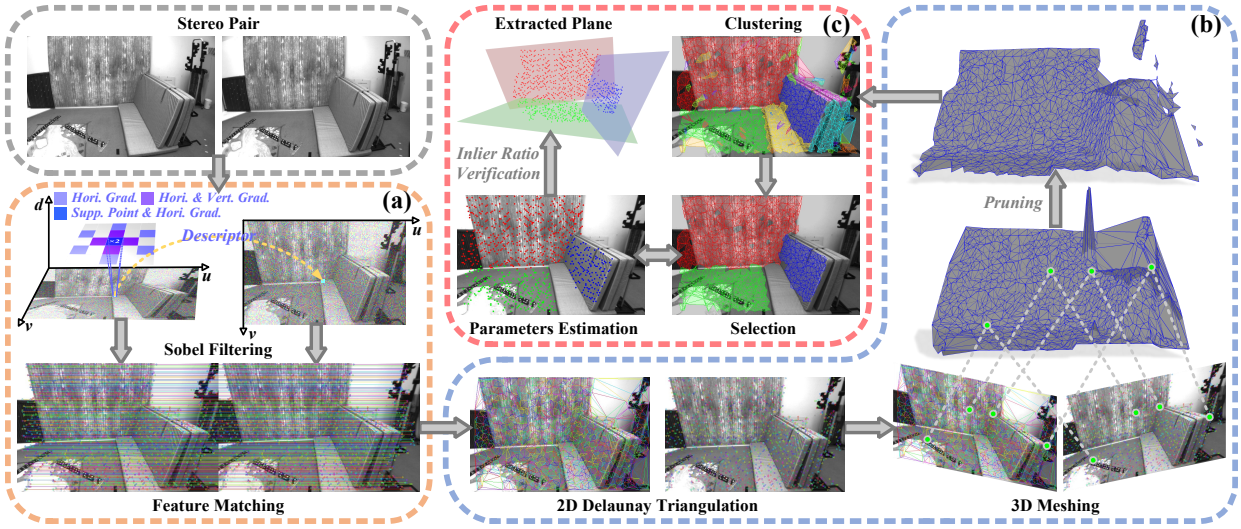


Fig. 2. Overview of the proposed robust stereo correspondences based plane extraction method. The orange dashed box shows the *Support Point Extraction* process, through which the 3D positions of the support points are obtained. The blue dashed box shows the *3D Mesh Generation* process, through which a 3D mesh is generated. The red dashed box shows the *Plane Extraction* process, through which the parameters of the main planes are estimated.

each pixel, forming a 16-dimensional descriptor $\mathbf{f}(u,v) = (G_{x_1},...,G_{x_{12}},G_{y_1},...,G_{y_4})^\top$, as shown in Fig. 2(a).

Feature Matching: Since the input images have been rectified, the stereo correspondences are restricted to the same row in both images. We first set the search range $d \in [d_{\min}, d_{\max}]$ and a grid with a fixed step size of 5 pixels. Then, for a Sobel feature $\mathbf{p}^l = (u,v)$ located on the grid in the left image of a stereo pair, feature matching is performed to search the pixel $\mathbf{p}^r$ with the highest response $E(d)$, along the v-row of the right image:

$$E(d) = \sum_{i,j\in\{-2,2\}} \left|\mathbf{f}^l(u+i,v+j) - \mathbf{f}^r(u-d+i,v+j)\right|, \tag{1}$$

where $\mathbf{f}^l(u+i,v+j)$ and $\mathbf{f}^r(u-d+i,v+j)$ are the descriptors of the corner points within the 5×5 neighborhood around $\mathbf{p}^l$ and $\mathbf{p}^r = (u-d,v)$ respectively.

To obtain reliable matching results, we rigorously perform left-right and disparity consistency checks, as well as a ratio test on stereo correspondences [13]. Once the robust stereo correspondences $\mathbf{p} = (u,v)$ and their disparity values d are obtained, they are selected as 2D support points for 3D mesh generation, whose 3D spatial coordinates $\mathbf{P}_f$ are computed through triangulation as follows:

$$\mathbf{P}_f = \begin{bmatrix} x & y & z \end{bmatrix}^\top = \begin{bmatrix} \frac{(u-c_x)z}{f_x} & \frac{(v-c_y)z}{f_y} & \frac{f_x b}{d} \end{bmatrix}^\top, \tag{2}$$

where (f_x, f_y) is the focal length, (c_x, c_y) is the principal point, and b is the baseline, all known from calibration.

B. 3D Mesh Generation

2D Delaunay Triangulation: Generating a 3D mesh directly from 3D spatial points is difficult. Therefore, we construct the 3D mesh by performing Delaunay triangulation [23] on the extracted 2D support points within the left image, yielding a set of triangles with support points as vertices.

3D Meshing: After 2D Delaunay triangulation, we utilize the 3D positions of support points to project the triangulation within the left image as a 3D mesh. Then, we employ the strategy described in [24] to prune the mesh in order to remove outliers by discarding triangles with side lengths exceeding a certain threshold, and triangles with excessively large aspect ratios or small acute angles in 3D, as illustrated in Fig. 2(b). Compared with methods that construct 3D mesh from sparse tracked keypoints [3], [15], our method constructs a 3D mesh with richer structural information, facilitating accurate single-frame plane extraction.

C. Plane Extraction

Clustering: The triangles in the mesh are clustered based on a region-growing strategy according to their surface normals and locality, which is achieved by performing a breadth-first graph search over the 3D mesh triangles. We start by randomly selecting an initial triangle and then add its neighboring triangles to a search queue $\mathcal{S}$. Next, a triangle is selected and removed from $\mathcal{S}$. We compute the inner product $\mathbf{n}_d$ of the unit normal vectors of the initial and selected triangles (with normals $\mathbf{n}_{\text{ini}}$ and $\mathbf{n}_{\text{sel}}$) to determine directional similarity:

$$\mathbf{n}_d = \left(\frac{\mathbf{n}_{\text{ini}}}{\|\mathbf{n}_{\text{ini}}\|}\right)^\top \left(\frac{\mathbf{n}_{\text{sel}}}{\|\mathbf{n}_{\text{sel}}\|}\right). \tag{3}$$

If $\mathbf{n}_d$ exceeds a threshold, indicating their directions are sufficiently aligned, we include the selected triangle in the current cluster and add its neighbors to $\mathcal{S}$. The above process is repeated until $\mathcal{S}$ is empty, indicating there are no more adjacent triangles with direction close to the initial triangle. Iteration continues until all triangles are clustered.

Selection: As shown in Fig. 2(c), the Clustering process results in numerous small triangle clusters. In narrow or pseudo-plane regions, clusters occupy a limited spatial extent. The planes extracted from these small clusters cannot be persistently tracked, and the accuracy of these planes is greatly affected by noise, which actually decreases the pose estimation accuracy. Therefore, we discard these clusters. In texture-rich regions where support points are densely distributed, triangles

formed by these points have a smaller span orthogonal to depth. This renders their normal vectors highly sensitive to depth variations, resulting in some triangles failing to meet clustering conditions, despite the fact that the absolute depths of the support points are accurate, attributed to the rich texture. To improve plane completeness and associate them with more point features, we merge smaller clusters into specific larger clusters that share the same plane, thereby introducing more comprehensive point-on-plane constraints to the SLAM system.

Given the above situations, we propose a cluster selection method based on adjacency relationships. We select the smallest cluster $\mathcal{M}$ and identify its most adjacent cluster, $\mathcal{K}_1$, and second-most adjacent cluster, $\mathcal{K}_2$. Triangles adjacent to $\mathcal{K}_1$ in $\mathcal{M}$ are counted as λ_1, and triangles adjacent to $\mathcal{K}_2$ are counted as λ_2. If the ratio λ_2/λ_1 is below a certain threshold, $\mathcal{M}$ is merged into $\mathcal{K}_1$. If the ratio exceeds the threshold, $\mathcal{M}$ is determined to be in a boundary region and discarded. If λ_1 equals 0, $\mathcal{M}$ is considered to be in an isolated plane or pseudo-plane region and also discarded. The above process is repeated until the size of $\mathcal{M}$ exceeds a certain threshold.

Plane Parameters Estimation: After Clustering and Selection, we obtain the clusters that can accurately reflect the main planar structures in the scene. Then, co-planar support points can be directly obtained from the triangle clusters. Furthermore, we employ RANSAC [25] on the co-planar support points to accurately calculate the plane parameters. If sufficient inliers are found, we compute the inlier ratio. Only if the ratio exceeds a certain threshold will we extract a plane from the inlier set. Besides, to avoid redundancy, a newly extracted plane is discarded if its parameters are sufficiently close to the planes already in the feature set.

III. Point-Plane-Based Stereo SLAM System

In this section, we propose a real-time point-plane-based stereo SLAM system built upon ORB-SLAM2 [26]. The core concept of our SLAM system is to leverage the expansive spatial characteristics of planar structures to lengthen feature tracks, thereby introducing additional constraints to further improve pose estimation. Here we mainly introduce the plane-related operations. For details on point-related operations, please refer to [26].

A. Optimization Formulation

The proposed SLAM system utilizes point and plane features, as well as planar regularities to formulate the optimization problem. We denote a set of camera poses, the union of non-planar and planar point features, plane features, planar point features as $\mathcal{T} = \{\mathbf{T}_i \in SE(3) | i = 1, ..., m\}$, $\mathcal{X} = \left\{\mathbf{X}_j \in \mathbb{R}^3 \middle| j = 1, ..., n\right\}$, $\mathcal{P} = \left\{\mathbf{\Pi}_k \in \mathbb{P}^3 \middle| k = 1, ..., p\right\}$, $\mathcal{Y} = \left\{\mathbf{Y}_l^k \in \mathbb{R}^3 \middle| k = 1, ..., p; l = 1, ..., q\right\}$ respectively, then the optimization problem can be formulated as:

$$\begin{aligned}\{\mathcal{T}^*, \mathcal{X}^*, \mathcal{P}^*, \mathcal{Y}^*\} = \underset{\mathcal{T},\mathcal{X},\mathcal{P},\mathcal{Y}}{\arg\min} \sum_{\mathbf{T}_i, \mathbf{X}_j} \rho_{ij}\left(\left\|\mathbf{e}_{\mathbf{X}_j}^{\mathbf{T}_i}\right\|_{\Sigma_{ij}}^2\right) \\ + \sum_{\mathbf{T}_i, \mathbf{\Pi}_k} \rho_{ik}\left(\left\|\mathbf{e}_{\mathbf{\Pi}_k}^{\mathbf{T}_i}\right\|_{\Sigma_{ik}}^2\right) + \sum_{\mathbf{\Pi}_k, \mathbf{Y}_l^k} \rho_{kl}\left(\left\|\mathbf{e}_{\mathbf{Y}_l^k}^{\mathbf{\Pi}_k}\right\|_{\Sigma_{kl}}^2\right),\end{aligned} \tag{4}$$

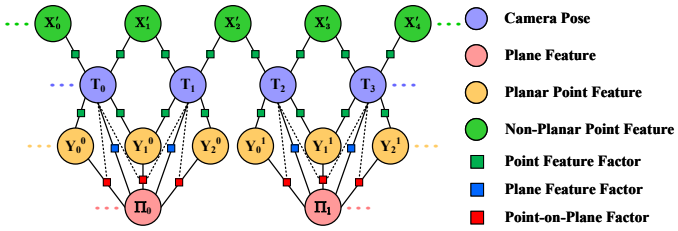


Fig. 3. Factor graph of the proposed point-plane-based stereo SLAM system. The variable nodes include camera pose $\mathbf{T}_i \in \mathcal{T}$, non-planar point feature $\mathbf{X}_j' \in \mathcal{X}$, plane feature $\mathbf{\Pi}_k \in \mathcal{P}$, and planar point feature $\mathbf{Y}_l^k \in \mathcal{Y}$. The factor nodes represent point and plane measurements, as well as the point-on-plane constraints between the associated planes and planar points respectively. In motion-only BA, the point-on-plane factor concurrently links $\mathbf{T}_i$, the observation of $\mathbf{\Pi}_k$, and associated $\mathbf{Y}_l^k$. In local BA, the point-on-plane factor links $\mathbf{\Pi}_k$ and associated $\mathbf{Y}_l^k$.

where $\mathbf{e}_{\mathbf{X}_j}^{\mathbf{T}_i}$, $\mathbf{e}_{\mathbf{\Pi}_k}^{\mathbf{T}_i}$ and $\mathbf{e}_{\mathbf{Y}_l^k}^{\mathbf{\Pi}_k}$ represent the camera-point, camera-plane measurement errors and point-on-plane errors respectively. $\| \mathbf{x} \|_{\Sigma}^2$ is the square of the Mahalanobis distance. Σ is the corresponding covariance matrix, which is set based on measurement uncertainty. $\rho(\cdot)$ is the robust Huber cost function. The factor graph of the proposed SLAM system is shown in Fig. 3.

B. Measurement Error

Camera-Point Error: A 2D point observation $\mathbf{x}_{i,j}$ in the image of camera $\mathbf{T}_i$ is matched with 3D point landmark $\mathbf{X}_j$. Then $\mathbf{X}_j$ is projected to image plane to calculate the reprojection error:

$$\mathbf{e}_{\mathbf{X}_j}^{\mathbf{T}_i} = \mathbf{x}_{i,j} - \operatorname{proj}\left(\mathbf{R}_i \mathbf{X}_j + \mathbf{t}_i\right), \tag{5}$$

where $\operatorname{proj}(\cdot) : \mathbb{R}^3 \rightarrow \Omega$ projects a 3D point in the camera frame to the image domain Ω, $\mathbf{R}_i \in SO(3)$ is the camera orientation, and $\mathbf{t}_i \in \mathbb{R}^3$ is the camera position. We employ the same method as ORB-SLAM2 [26] to determine its covariance matrix.

Camera-Plane Error: To construct camera-plane error, it is necessary to associate the plane observations with the plane landmarks in the map. We employ the method of [8] to associate plane observations with landmarks if the normal vector difference and the point-to-plane distance (the minimum distance from the support points on the plane landmark to the plane observation in the world frame) between them are less than certain thresholds.

Besides, since a 3D plane has only three degrees of freedom, the representation $\mathbf{\Pi} = \left(\mathbf{n}^\top, d\right)^\top$, where $\mathbf{n} = (n_x, n_y, n_z)^\top$, is an over-parameterization, which requires extra constraints to ensure that the normal vector of the plane maintains unit length during optimization. Therefore, we adopt the method in [11] using the minimal parameterization of plane $q(\mathbf{\Pi}) = (\phi, \psi, d)$:

$$q(\mathbf{\Pi}) = \left(\phi = \arctan\left(\frac{n_y}{n_x}\right), \psi = \arcsin\left(n_z\right), d\right), \tag{6}$$

where ϕ and ψ are the azimuth and elevation angles of the normal vector respectively. Then, the reprojection error using the minimal parameterization is defined as follows:

$$\mathbf{e}_{\mathbf{\Pi}_k}^{\mathbf{T}_i} = q\left(\boldsymbol{\pi}_{i,k}\right) - q\left(\mathbf{T}_i^{-\top} \mathbf{\Pi}_k\right), \tag{7}$$

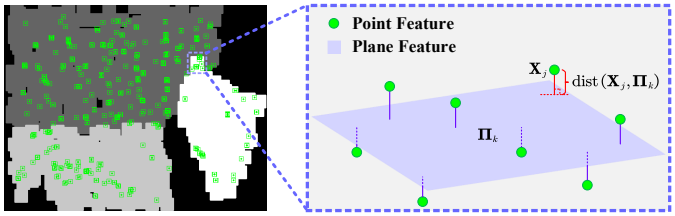


Fig. 4. Illustration of Point and Plane Features Association. The left image shows a mask image generated from the inlier sets of different plane features $\mathbf{\Pi}_k$, associating point features $\mathbf{X}_j$ with corresponding $\mathbf{\Pi}_k$ based on pixel coordinates. The right image further calculates the distance from each $\mathbf{X}_j$ to its initially associated $\mathbf{\Pi}_k$, eliminating the associations between $\mathbf{X}_j$ and $\mathbf{\Pi}_k$ where the distance exceeds a certain threshold.

where $\boldsymbol{\pi}_{i,k} \in \mathbb{P}^3$ is the observation of landmark $\mathbf{\Pi}_k$ in the image of camera $\mathbf{T}_i$. Assuming plane observation follows Gaussian distribution [21], the standard deviations of angle and distance are set at 2° and 1 cm, respectively, in our experiments.

Notably, new plane landmarks are created using planes extracted from keyframes that do not match with any plane landmarks in the map and are initially marked as invalid due to their potential unreliability [1]. Plane landmarks are reclassified as valid only when observed by more than three keyframes, and only valid landmarks are tracked for pose estimation. Furthermore, valid plane landmarks will be reclassified as invalid when observed by fewer than two keyframes due to excessive reprojection errors.

Point-on-Plane Error: To construct the point-on-plane error, it is necessary to associate the point features with their corresponding plane features. First, we obtain 2D inlier sets $\{\mathcal{I}_1, \mathcal{I}_2 ..., \mathcal{I}_p\}$ after applying RANSAC, where $\mathcal{I}_k = \left\{\left(u_1^k, v_1^k\right), \left(u_2^k, v_2^k\right), ..., \left(u_{k_s}^k, v_{k_s}^k\right)\right\}$ represents the inlier set belonging to the k-th plane feature $\mathbf{\Pi}_k$ with k_s being the number of inliers, and p represents the number of plane features. Then, we create a mask image $\mathbf{M}$ of the same size as the original image and set the pixel values to 0. Fianlly, we generate a rectangular mask area of size $w \times h$ around each inlier $\left(u_e^k, v_e^k\right)$. The grayscale value of the mask area generated from the k-th inlier set is $\mathcal{G}_k$, where $\mathcal{G} = \{g_1, g_2, ..., g_p\}$:

$$\mathbf{M}(u, v) = \mathcal{G}_k, \; if \; \left|u - u_e^k\right| \leqslant w/2 \wedge \left|v - v_e^k\right| \leqslant h/2, \quad (8)$$

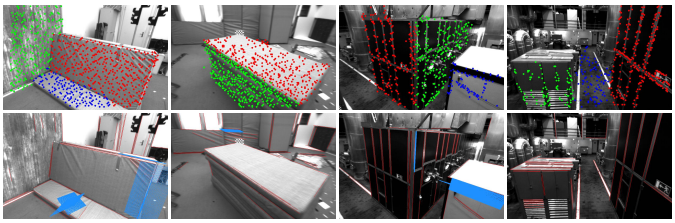
where $k \in [1, p]$, $e \in [1, k_s]$. The grayscale value of $\mathbf{M}(u, v)$ is updated only if its current value is 0 for each pixel (u, v). After obtaining the mask image, we associate the point feature $\mathbf{Y}_l^k$ with the plane feature $\mathbf{\Pi}_k$ as shown in Fig. 4, when satisfying the following conditions:

- The observation of $\mathbf{Y}_l^k$ is within the mask area generated from the inliers of $\mathbf{\Pi}_k$ in 2D.
- The distance between $\mathbf{Y}_l^k$ and $\mathbf{\Pi}_k$ is less than a certain threshold in 3D.

If both $\mathbf{\Pi}_k$ and $\mathbf{Y}_l^k$ are valid features, a point-on-plane constraint is established between them. Then, we construct the point-on-plane errors $\mathbf{e}_{m_{\mathbf{Y}_l^k}^{\mathbf{\Pi}_k}}$ in motion-only BA and $\mathbf{e}_{l_{\mathbf{Y}_l^k}^{\mathbf{\Pi}_k}}$ in local BA, respectively as follows:

$$\mathbf{e}_{m_{\mathbf{Y}_l^k}^{\mathbf{\Pi}_k}} = \mathbf{n}_{\boldsymbol{\pi}_{i,k}}^{\top}\left(\mathbf{R}_i \mathbf{Y}_l^k + \mathbf{t}_i\right) - d_{\boldsymbol{\pi}_{i,k}}, \quad (9)$$

$$\mathbf{e}_{l_{\mathbf{Y}_l^k}^{\mathbf{\Pi}_k}} = \mathbf{n}_{\mathbf{\Pi}_k}^{\top} \mathbf{Y}_l^k - d_{\mathbf{\Pi}_k}. \quad (10)$$



(a) Vicon Room Scene (b) Machine Hall Scene

Fig. 5. Plane extraction results of the proposed plane extraction algorithm and SP-SLAM in the Vicon Room and Machine Hall scenes. For each scene, the top row displays the results from the proposed algorithm, and the bottom row displays the results from SP-SLAM.

During the motion-only BA, landmarks are fixed and only the camera pose is optimized. During the local BA, a local window of point and plane landmarks is jointly optimized through mutual constraints. Assuming the distance error follows Gaussian distribution [3], the standard deviation is set at 1.5 cm in our experiments.

IV. Experiments

In this section, we evaluate the proposed plane extraction algorithm on the EuRoC dataset [27], comparing it with a state-of-the-art (SOTA) point-plane-based stereo SLAM system (SP-SLAM [1]). Furthermore, we evaluate the proposed SLAM system on the EuRoC dataset [27] and the KITTI dataset [28]. Two SOTA SLAM systems (ORB-SLAM2 [26], SP-SLAM [1]) are selected for comparison. ORB-SLAM2 is a point-based SLAM system with a stereo camera implementation. SP-SLAM is a point-plane-based stereo SLAM system that computes plane features from two intersecting lines in stereo images. It is also built upon ORB-SLAM2 and achieves SOTA performance. For all the systems, we disable the loop closure detection module to only compare their odometry performances. We use the root mean squared error (RMSE) of the absolute trajectory error (ATE) as the evaluation metric. To reduce random errors, we run each system five times and report the median. All experiments run on a desktop with AMD Ryzen 7 4800H CPU (with @2.9GHz), and 32GB RAM.

A. Plane Extraction Evaluation

We compare the proposed plane extraction algorithm with the plane extraction module of SP-SLAM. Since SP-SLAM can extract only a very limited number of planes from the KITTI dataset, we evaluate the algorithms on the EuRoC dataset without considering the KITTI dataset. Notably, to isolate the effect of the plane extraction module, we use the ground truth poses in the experiments.

To conduct a comprehensive evaluation, we adopt the following metrics: valid plane number, valid plane ratio, reprojection error, observation count, and error term number. The valid plane number is the total count of plane landmarks observed by a minimum of three keyframes. The valid plane ratio is the proportion of valid plane landmarks to all plane landmarks. The reprojection error, calculated by Eq. (7), is the average matching error between plane observations and landmarks, using the squared norm as the metric. The observation count is the average number of times that valid plane landmarks are observed by keyframes. The error term number is the

TABLE I
PLANE EXTRACTION PERFORMANCE COMPARISON OF THE PROPOSED METHOD AND SP-SLAM [1] (THE UPWARD ARROW MEANS THE LARGER THE METRIC, THE BETTER, WHILE THE DOWNWARD ARROW MEANS THE SMALLER THE METRIC, THE BETTER).

Sequence	Valid Plane Number ↑		Valid Plane Ratio ↑		Reprojection Error ↓		Observation Count ↑		Error Term Number ↑	
	SP-SLAM	Ours	SP-SLAM	Ours	SP-SLAM	Ours	SP-SLAM	Ours	SP-SLAM	Ours
V1_01_easy	15	**27**	0.054	**0.284**	0.021	**0.014**	6.000	**14.037**	7409	**27724**
V1_02_medium	14	**35**	0.032	**0.205**	0.019	**0.016**	6.286	**11.543**	2067	**15823**
V1_03_difficult	6	**43**	0.009	**0.128**	0.025	**0.020**	8.167	**10.651**	2129	**19860**
V2_01_easy	1	**21**	0.010	**0.099**	**0.011**	0.020	5.000	**6.714**	76	**7223**
V2_02_medium	6	**42**	0.023	**0.090**	0.018	**0.013**	5.167	**8.905**	1034	**20858**
MH_01_easy	15	**35**	0.024	**0.118**	0.013	**0.010**	12.133	**17.933**	6707	**64817**
MH_02_easy	12	**30**	0.026	**0.103**	0.015	**0.013**	13.417	**17.833**	7998	**54348**
MH_03_medium	5	**25**	0.009	**0.092**	0.015	**0.011**	15.200	**20.600**	1763	**41045**
MH_04_difficult	4	**8**	0.012	**0.053**	0.016	**0.014**	9.500	**12.625**	1178	**4755**
MH_05_difficult	4	**11**	0.009	**0.073**	0.035	**0.015**	9.500	**16.636**	1792	**9979**

total count of reprojection error terms involving valid plane landmarks. These metrics evaluate the extracted planes from different perspectives: number, accuracy, and reliability, which are crucial for enhancing the performance of SLAM systems.

Table I shows the results of two plane extraction algorithms on the EuRoC dataset. Our method outperforms SP-SLAM in valid plane number and ratio on all sequences, demonstrating its effectiveness and efficiency in extracting reliable planes, which benefits from the full exploitation of structural information and the targeted plane selection strategy. In contrast, as shown in Fig. 5, SP-SLAM has two main limits: few intersecting lines meeting plane extraction conditions, and a tendency to extract false planes in cluttered scenes, resulting in a lower valid plane number and ratio. Besides, our method exhibits smaller reprojection errors on most sequences, indicating that the planes extracted by our method are more accurate, considering that under the use of ground truth poses, the magnitude of errors is primarily associated with plane accuracy. This benefits from the robustly matched support points for plane extraction, making planes less susceptible to noise, as well as the strict outlier rejection mechanisms. In contrast, SP-SLAM, which computes plane parameters using endpoints of intersecting lines, is significantly more noise-sensitive, as shown in Fig. 5(a).

In addition, the increased number and enhanced accuracy of valid planes lead to plane landmarks being observed by more keyframes, thereby constructing more error terms. Therefore, our plane extraction algorithm achieves higher observation count and error terms on all sequences, indicating that the planes extracted by our algorithm can be tracked over longer distances and introduce sufficient constraints into the system. In summary, our algorithm extracts more planes stable across multiple frames and trackable over longer distances, with more accurate parameters. These planes provide more comprehensive and accurate constraints across a broader temporal and spatial range for pose estimation.

B. Point-Plane-Based SLAM on EuRoC Dataset

The EuRoC dataset [27] comprises 11 stereo sequences captured by a micro aerial vehicle (MAV) flying around two Vicon rooms and an industrial machine hall. These scenes contain walls, floors, and some other planar objects. Table II shows the RMSE ATE results on the EuRoC dataset, highlighting the minimum error of each sequence in bold. Compared with ORB-SLAM2, the proposed system achieves higher accuracy on all sequences except MH_01_easy, and SP-SLAM also outperforms ORB-SLAM2 on sequences rich in intersecting lines suitable for plane extraction, demonstrating the effectiveness of appropriate plane features for improving pose estimation. The better performance of ORB-SLAM2 on the MH_01_easy sequence benefits from the accurate point features, attributed to the simple motion pattern and stable imaging conditions of this sequence, where adopting less accurate planar regularities instead degrades performance. A map built by our SLAM system is shown in Fig. 1, which accurately reflects the planar structures of the real scene, where planes from walls and ground are matched even at significant distances, further verifying the effectiveness of our plane extraction algorithm.

TABLE II
PERFORMANCE COMPARISON ON THE EUROC DATASET (RMSE ATE IN METER).

Sequence	ORB-SLAM2	SP-SLAM	w/o PR & POP	w/o PR	w/o POP	Ours
V1_01_easy	0.0865	0.0869	0.0868	0.0862	0.0863	**0.0861**
V1_02_medium	0.1037	0.0956	0.1058	0.0833	0.0767	**0.0698**
V1_03_difficult	0.2654	0.2275	0.2579	0.2288	0.2148	**0.1851**
V2_01_easy	0.0680	0.0719	0.0695	**0.0672**	0.0771	0.0677
V2_02_medium	0.0934	0.1073	0.0919	0.0953	0.1046	**0.0824**
MH_01_easy	**0.0353**	0.0368	0.0356	0.0357	0.0383	0.0365
MH_02_easy	0.0358	0.0354	0.0361	**0.0349**	0.0362	**0.0349**
MH_03_medium	0.0411	0.0372	0.0402	**0.0364**	0.0375	0.0370
MH_04_difficult	0.1037	0.0979	0.1084	0.0952	0.0945	**0.0918**
MH_05_difficult	0.0607	0.0796	0.0658	0.0682	0.0686	**0.0457**
Average	0.0894	0.0876	0.0898	0.0831	0.0835	**0.0737**

Besides, our SLAM system outperforms SP-SLAM on all sequences. To get more insight into the effectiveness of our system, we conduct an ablation study to evaluate our plane extraction and exploitation methods. For simplicity and comparison, we denote our SLAM system without point-on-plane constraints as w/o POP, without plane reprojection constraints as w/o PR, and without both plane reprojection constraints and point-on-plane constraints as w/o PR & POP. w/o POP differs from SP-SLAM only in the plane extraction algorithm, w/o PR only adds point-on-plane constraints compared with ORB-SLAM2, and w/o PR & POP does not utilize any plane constraints. As shown in Table II, w/o POP outperforms SP-SLAM by 1-2 cm in accuracy, attributed to the enhanced plane extraction algorithm detailed in Sec. IV-A. In addition, w/o PR achieves more accurate trajectories than ORB-SLAM2 on most sequences, particularly in challenging conditions with illumination changes or motion blur such as V1_03_difficult and MH_04_difficult. This is because, in these sequences, planes with smaller errors enhance the accuracy

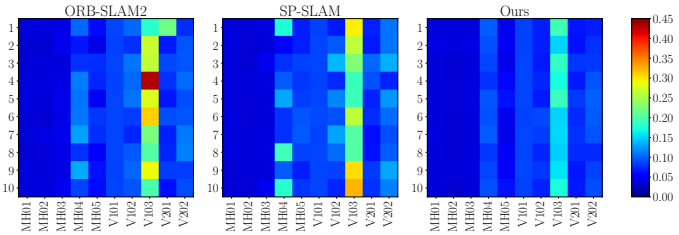


Fig. 6. Colored squares represent the RMSE ATE for ten different executions on each sequence of the EuRoC dataset.

TABLE III
PERFORMANCE COMPARISON ON THE KITTI DATASET (RMSE ATE IN METER).

Sequence	ORB-SLAM2	SP-SLAM	w/o PR & POP	w/o PR	w/o POP	Ours
00	3.958	4.017	3.994	3.938	3.773	**3.754**
01	12.446	12.477	12.419	11.515	10.898	**10.574**
02	9.362	9.100	9.294	9.209	8.767	**8.728**
03	0.649	0.677	0.652	**0.633**	0.671	0.654
04	**0.177**	0.219	0.182	0.188	0.203	0.199
05	2.129	2.074	2.103	2.068	2.065	**2.056**
06	2.368	2.358	2.377	2.194	2.194	**2.152**
07	1.519	1.500	1.532	1.397	1.361	**1.309**
08	3.647	**3.563**	3.641	3.638	3.620	3.625
09	3.276	3.397	3.295	3.288	3.252	**3.200**
10	1.087	1.090	1.092	1.110	1.082	**1.073**
Average	3.693	3.679	3.689	3.562	3.444	**3.395**

of point landmarks via Eq. (10), and introduce additional constraints in pose optimization through Eq. (9), indirectly and directly enhancing pose estimation accuracy. Moreover, the performance of w/o PR & POP is close to that of ORB-SLAM2, which proves that plane constraints are indeed the source of system performance improvement. Notably, Table II shows that utilizing both plane reprojection and point-on-plane constraints in our SLAM system results in a significant 1-5 cm improvement in accuracy over w/o PR and w/o POP. This enhancement, particularly evident in challenging sequences such as V1_03_difficult and MH_05_difficult, demonstrates that there is a mutual enhancement effect between geometric feature (point and plane) and across-feature (point-on-plane) constraints, making the associated point and plane achieve reduced error through mutual constraints, and consequently leading to more accurate pose estimation.

To investigate the impact of plane features on the robustness of the systems, we present colored squares representing the RMSE ATE for ten different executions on each sequence of the EuRoC dataset, as shown in Fig. 6. All systems show high robustness on easy sequences such as MH_01_easy-MH_03_medium and V1_01_easy, but on more challenging sequences such as MH_04_difficult and V1_03_difficult, SP-SLAM and ORB-SLAM2 exhibit considerable variability, due to the significant errors in point features and intersecting lines. Nevertheless, SP-SLAM shows more consistent errors than ORB-SLAM2 overall, indicating that appropriate plane feature integration improves systems robustness. Notably, our SLAM system exhibits increased robustness on all sequences, benefiting from the advanced plane extraction algorithm and full exploitation of plane features.

C. Point-Plane-Based SLAM on KITTI Dataset

The KITTI dataset [28] comprises stereo sequences from urban, rural, and highway scenarios, rich in long-tracked planes

TABLE IV
EXECUTION TIME COMPARISON (IN MILLISECONDS).

Set.	System	ORB-SLAM2		SP-SLAM		Ours	
	Sequence	V202	00	V202	00	V202	00
Trac.	Point Ext.	18.6	25.1	18.3	25.1	18.5	25.2
	Line Ext.	-	-	22.9	39.1	-	-
	Plane Ext.	-	-	0.01	0.01	16.0	17.1
	Associate	-	-	-	-	0.1	0.2
	Feat. Proc.	18.6	25.1	23.0	39.1	18.7	25.5
	Pose Pred.	6.9	7.8	7.6	8.1	9.5	10.6
	Total	25.6	33.0	30.7	47.2	28.3	36.2
Map.	LBA	63.1	73.4	68.5	74.2	69.7	80.6

TABLE V
EXECUTION TIME OF EACH COMPONENT OF THE PROPOSED PLANE EXTRACTION ALGORITHM (IN MILLISECONDS).

Sequence	SPE	MG	Cluster	Select	PPE	Total
V202	14.7	0.3	0.5	0.4	0.1	16.0
00	15.8	0.3	0.5	0.4	0.1	17.1

such as roads and buildings. Table III shows that our SLAM system outperforms ORB-SLAM2 on most sequences, except for Seq. 03 and 04, further demonstrating the effectiveness of planar regularities in improving pose estimation accuracy. Our SLAM system achieves lower accuracy on Seq. 03 and 04 due to the substantial presence of cluttered trees alongside roads in these rural sequences, leading to false plane extraction and inaccurate planar regularities.

Besides, the performance of SP-SLAM and ORB-SLAM2 is comparable on the KITTI dataset, due to the limited valid planes extracted by SP-SLAM. An ablation study with the same settings as Sec. IV-B shows that the performance of w/o PR & POP is close to that of ORB-SLAM2, w/o PR and w/o POP outperform two SOTA SLAM systems on most sequences, and removing any plane constraint from our SLAM system leads to a significant reduction in trajectory accuracy, further demonstrating the effectiveness of our plane extraction and exploitation method, as well as the mutual enhancement between two types of constraints, in outdoor environments.

D. Runtime Evaluation

Table IV compares the average runtime of ORB-SLAM2, SP-SLAM, and the proposed SLAM system on the EuRoC V2_02_medium and KITTI Seq. 00. Similar to SP-SLAM, we employ parallel CPU threads to extract different features, therefore the feature processing time is approximately equal to the longest individual feature extraction time. Benefiting from our efficient plane extraction module, the feature processing time of our SLAM system is close to that of ORB-SLAM2, approximately equal to the point extraction time. Compared with our efficient plane extraction module, with the runtime of different components (Support Point Extraction, 3D Mesh Generation, Clustering, Selection, Plane Parameters Estimation abbreviated as SPE, MG, Cluster, Select and PPE respectively) detailed in Table V, the line extraction module of SP-SLAM is more time-consuming, and many extracted lines fail to satisfy the plane extraction conditions, as shown in Fig. 5. Moreover, the local optimization time of SP-SLAM is close to ORB-SLAM2 on the KITTI dataset, as it extracts few valid planes, resulting in a marginal contribution to pose

optimization. However, it consumes substantial computational resources to extract invalid lines, rendering its plane extraction and exploitation less efficient than our SLAM system.

V. CONCLUSIONS

This paper presents a novel point-plane-based stereo SLAM system, featuring an advanced stereo plane extraction algorithm for real-time extraction of main planes from robust stereo correspondences, and a new optimization formulation for full exploitation of plane features through mutually enhancing constraints, significantly improving the performance of SLAM system. All the novelties contribute to an accurate and robust SLAM system, which can also build a geometric plane map that accurately reflects real scene structures. The experimental results on the EuRoC dataset show that the proposed SLAM system extracts, on average, 3.3 times more valid planes with smaller errors, while speeding up feature processing by up to 34.7% compared with the SOTA stereo point-plane-based system [1], achieving significant improvement in number, accuracy, reliability, and efficiency. The results of an ablation study on two public datasets show that the proposed SLAM system fully exploits plane constraints with higher efficiency in both indoor and outdoor scenes, significantly improving accuracy and robustness compared with [1], and further demonstrate the mutual enhancement between the two types of constraints.

However, the proposed method may extract planes that are inaccurate or even wrong in cluttered scenes, compromising the system performance. In future work, we hope to enhance the plane extraction algorithm with more rigorous outlier rejection and plane verification mechanisms, calculate plane reliability and set optimization weights, as well as integrate semantic information to mitigate the limitation. We also plan to integrate the loop-closure module into our SLAM system to further improve its consistency, and explore efficient dense reconstruction utilizing plane features with a stereo camera.

REFERENCES

[1] X. Zhang, W. Wang, X. Qi, and Z. Liao, "Stereo plane slam based on intersecting lines," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6566–6572, IEEE, 2021.

[2] C. Chen, P. Geneva, Y. Peng, W. Lee, and G. Huang, "Monocular visual-inertial odometry with planar regularities," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6224–6231, IEEE, 2023.

[3] X. Li, Y. He, J. Lin, and X. Liu, "Leveraging planar regularities for point line visual-inertial odometry," in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 5120–5127, IEEE, 2020.

[4] A. Dai, G. Lund, and G. Gao, "Planeslam: Plane-based lidar slam for motion planning in structured 3d environments," *arXiv preprint arXiv:2209.08248*, 2022.

[5] F. Shu, J. Wang, A. Pagani, and D. Stricker, "Structure plp-slam: Efficient sparse mapping and localization using point, line and plane for monocular, rgb-d and stereo cameras," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2105–2112, IEEE, 2023.

[6] H. Wei, F. Tang, Z. Xu, and Y. Wu, "Structural regularity aided visual-inertial odometry with novel coordinate alignment and line triangulation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10613–10620, 2022.

[7] L. Zhou, D. Koppel, and M. Kaess, "Lidar slam with plane adjustment for indoor environment," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7073–7080, 2021.

[8] X. Zhang, W. Wang, X. Qi, Z. Liao, and R. Wei, "Point-plane slam using supposed planes for indoor environments," *Sensors*, vol. 19, no. 17, p. 3795, 2019.

[9] T. Shan and B. Englot, "Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4758–4765, IEEE, 2018.

[10] A. J. Trevor, S. Gedikli, R. B. Rusu, and H. I. Christensen, "Efficient organized point cloud segmentation with connected components," *Semantic Perception Mapping and Exploration (SPME)*, vol. 10, no. 6, pp. 251–257, 2013.

[11] L. Ma, C. Kerl, J. Stückler, and D. Cremers, "Cpa-slam: Consistent plane-model alignment for direct rgb-d slam," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1285–1291, IEEE, 2016.

[12] C. Arndt, R. Sabzevari, and J. Civera, "From points to planes-adding planar constraints to monocular slam factor graphs," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4917–4922, IEEE, 2020.

[13] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International journal of computer vision*, vol. 47, pp. 7–42, 2002.

[14] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: an open-source library for real-time metric-semantic localization and mapping," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1689–1696, IEEE, 2020.

[15] A. Rosinol, T. Sattler, M. Pollefeys, and L. Carlone, "Incremental visual-inertial 3d mesh generation with structural regularities," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8220–8226, IEEE, 2019.

[16] X. Li, Y. Li, E. P. Örnek, J. Lin, and F. Tombari, "Co-planar parametrization for stereo-slam and visual-inertial odometry," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6972–6979, 2020.

[17] M. Usayiwevu, F. Sukkar, and T. Vidal-Calleja, "Probabilistic plane extraction and modeling for active visual-inertial mapping," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10601–10607, IEEE, 2023.

[18] D. Chen, S. Wang, W. Xie, S. Zhai, N. Wang, H. Bao, and G. Zhang, "Vip-slam: An efficient tightly-coupled rgb-d visual inertial planar slam," in *2022 International Conference on Robotics and Automation (ICRA)*, pp. 5615–5621, IEEE, 2022.

[19] X. Wang, M. Christie, and E. Marchand, "Tt-slam: Dense monocular slam for planar environments," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11690–11696, IEEE, 2021.

[20] K. Ram, C. Kharyal, S. S. Harithas, and K. M. Krishna, "Rp-vio: robust plane-based visual-inertial odometry for dynamic environments," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 9198–9205, IEEE, 2021.

[21] Y. Li, R. Yunus, N. Brasch, N. Navab, and F. Tombari, "Rgb-d slam with structural regularities," in *2021 IEEE international conference on Robotics and automation (ICRA)*, pp. 11581–11587, IEEE, 2021.

[22] C. Arndt, R. Sabzevari, and J. Civera, "Do planar constraints improve camera pose estimation in monocular slam?," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2221–2230, 2023.

[23] L. P. Chew, "Constrained delaunay triangulations," in *Proceedings of the third annual symposium on Computational geometry*, pp. 215–222, 1987.

[24] A. Rosinol, "Densifying sparse vio: a mesh-based approach using structural regularities," Master's thesis, ETH Zurich; Massachusetts Institute of Technology, 2018.

[25] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[26] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[27] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The euroc micro aerial vehicle datasets," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.

[28] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE conference on computer vision and pattern recognition*, pp. 3354–3361, IEEE, 2012.