



Article A Robust Planar Marker-Based Visual SLAM

Zhoubo Wang¹, Zhenhai Zhang^{1,*}, Wei Zhu^{1,*}, Xuehai Hu¹, Hongbin Deng¹, Guang He¹ and Xiao Kang²

- ¹ School of Mechatronical Engineering, Beijing Institute of Technology, Beijing 100081, China
- ² UV Center, China North Vehicle Research Institute, Beijing 100072, China
- * Correspondence: zhzhang@bit.edu.cn (Z.Z.); wei.zhu@bit.edu.cn (W.Z.)

Abstract: Many visual SLAM systems are generally solved using natural landmarks or optical flow. However, due to textureless areas, illumination change or motion blur, they often acquire poor camera poses or even fail to track. Additionally, they cannot obtain camera poses with a metric scale in the monocular case. In some cases (such as when calibrating the extrinsic parameters of camera-IMU), we prefer to sacrifice the flexibility of such methods to improve accuracy and robustness by using artificial landmarks. This paper proposes enhancements to the traditional SPM-SLAM, which is a system that aims to build a map of markers and simultaneously localize the camera pose. By placing the markers in the surrounding environment, the system can run stably and obtain accurate camera poses. To improve robustness and accuracy in the case of rotational movements, we improve the initialization, keyframes insertion and relocalization. Additionally, we propose a novel method to estimate marker poses from a set of images to solve the problem of planar-marker pose ambiguity. Compared with the state-of-art, the experiments show that our system achieves better accuracy in most public sequences and is more robust than SPM-SLAM under rotational movements. Finally, the open-source code is publicly available and can be found at GitHub.

Keywords: visual SLAM; planar markers; pose ambiguity

1. Introduction

Multi-sensor fusion has attracted the interest of SLAM researchers in the field of autonomous robots. VINS [1], ORB-SLAM3 [2], and Smart Markers [3] are excellent visual–inertial systems. However, the precise extrinsic parameters of camera-IMU are a pre-requisite for these systems. Although some systems (such as VINS and VI-ORB-SLAM [4]) provide online calibration methods, they often fail to calibrate the extrinsic parameters of a camera-IMU mounted on vehicles. This is because the vehicle generally drives on a flat surface and does not generate enough excitation. In this case, accurate camera poses are beneficial, allowing for us to calibrate the external parameters of camera-IMU using the hand–eye method [5]. Many outstanding visual SLAM systems can be used to obtain camera poses, such as ORB-SLAM2 [6], LSD-SLAM [7], and LDSO [8]. However, these methods, based on natural features or optical flow, could fail under illumination conditions, or instances with high dynamics, vigorous rotation, or a low-texture environment [9]. Additionally, they cannot obtain camera poses with a metric scale in the monocular case. Therefore, we prefer to sacrifice the flexibility of such methods to improve accuracy and robustness using artificial planar landmarks.

The planar marker is an easy-to-obtain artificial mark. We can easily print the ArUco [10,11] or AprilTag [12] markers in our laboratory. Therefore, planar-marker-based SfM or SLAM are very popular [13–17]. Although marker-based SLAM is much simpler than feature-based SLAM, a challenging problem is that the planar marker contains ambiguity. The pose is the relative pose between the camera and the marker. Due to the noise in the four corners of the detection marker, two different solutions will be estimated in practice, and ambiguity only occurs in the rotational components of the pose [18–20]. Generally, we use the 'infinitesimal plane-based pose estimation' (IPPE) method [20] to obtain the two



Citation: Wang, Z.; Zhang, Z.; Zhu, W.; Hu, X.; Deng, H.; He, G.; Kang, X. A Robust Planar Marker-Based Visual SLAM. *Sensors* 2023, 23, 917. https://doi.org/ 10.3390/s23020917

Academic Editor: Srikanth Saripalli

Received: 11 December 2022 Revised: 1 January 2023 Accepted: 11 January 2023 Published: 13 January 2023



Copyright: © 2023 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/). solutions, which include the correct one. Figure 1 shows two solutions ξ' (the red bounding box) and ξ'' (the green bounding box), which are returned by IPPE method for a planar marker with ID 205. The red bounding box is the correct solution. The literature [14,15] has proposed several means to distinguish the correct solution. Nevertheless, these methods are still not robust enough and have difficulties in rotational movements.



Figure 1. (a): A detected marker with ID 205 from an image. (b): The two pose solutions, ξ' (red bounding box) and ξ'' (green bounding box), returned by IPPE. The blue bounding box is the camera pose.

To avoid or alleviate the difficulties in rotational movements, we propose enhancements to the traditional SPM-SLAM system [15]. We present an improved version of this existing algorithm. We mainly improve four steps in SPM-SLAM. First, in the one-frame initialization, the correct maker pose is only chosen by IPPE if the marker is within a suitable view of the camera and the *ratio* from Equation (4) is lower than the threshold (Section 4.1). The two-frame initialization is successful if the distance or angle between two frames is greater than the corresponding threshold (Section 4.1). This makes it easier for the initialization to succeed under rotational movements. Second, the current frame can be inserted into the map as a keyframe if the angle between the current frame and the reference keyframe is greater than the threshold. This ensures that our system inserts enough keyframes under the rotation movements to continue tracking (Section 4.2). Third, we present a new approach to select the best marker pose from a set of images by minimizing the M2M error (Section 4.3). Finally, during the relocalization process, when one marker is detected, its correct pose is chosen using the same method as the one-frame initialization approach. When multiple markers are detected, their correct poses are chosen by minimizing the M2M error (Section 4.4). We tested our system on the public dataset and our dataset. Compared with SPM-SLAM [15], UcoSLAM [16], and TagSLAM [17], our system achieves better accuracy in most public sequences, and the speed of our system is much faster than that of UcoSLAM and TagSLAM. The results of our dataset show that our method is more robust and obtains more planar markers in the map than SPM-SLAM under rotational movements. Additionally, the open-source code is available on GitHub.

2. Related Research

As our work is dedicated to the planar marker-based SLAM system, a review of the natural feature-based SLAM literature is beyond the scope of this paper. Both Ref. [9] and Ref. [21] contain detailed reviews of natural feature-based SLAM systems. This section will focus on the marker-based SLAM and the problem of marker pose ambiguity.

SPM-SLAM [15] is the first real-time marker-based SLAM system. It is able to deal with ambiguity problems and operate in large indoor environments. The SLAM system sequentially runs the tracking, mapping, and loop-closing processes in a single thread and achieves a good performance in most environments. However, our experiments

(Section 5.2) indicated that SPM-SLAM often failed to locate in certain situations where the camera's motion included rotational movements. UcoSLAM [16] is an extended version of SPM-SLAM and fuses the keypoints with squared fiducial markers. When disabling keypoints, it is similar to SPM-SLAM. However, it needs more computational resources and the addition of keypoints did not significantly improve SPM-SLAM's tracking accuracy in our experiments.TagSLAM [17] is also a visual SLAM with AprilTag fiducial markers. It considers every frame to be a keyframe and relies on iSAM2 [22], which uses a factor graph to represent the pose optimization problems. However, TagSLAM cannot run in real-time unless a trusted map of tag poses is already available, because the graph grows over time and the CPU load increases. Our experiments indicated that TagSLAM is unstable. MarkerMapper [14] is an offline method that obtains a map of fiducial markers and then obtains camera poses. It has several limitations. The most serious one is that the proposed method cannot run in real-time.

Additionally, the *ratio* test [20] is a general method to resolve the planar-marker's pose ambiguity. Marker-based SfM [14], SPM-SLAM [15], and UcoSLAM [16] also apply this method. The *ratio* test means that if the *ratio* from Equation (4) is below a certain threshold, the IPPE solution with lower reprojection errors is the correct one. However, this was proved to not always be correct. Unlike the *ratio* test, which used one frame to solve marker pose ambiguity, S. -F. Ch'ng [23] exploited multi-view constraints for disambiguation. This method can efficiently choose the correct pose from two ambiguous poses by formulating a clique-constrained rotation averaging problem and a maximum weighted clique problem. However, the method consumes a high amount of computing resources and is not suitable for SLAM running in real-time.

3. System Overview

The enhanced marker-based SLAM is a monocular SLAM system. It is an accurate and real-time system used to estimate camera poses and create a map of planar markers. To describe the improvements that we have made, Section 3.1 explains some notations that were employed in this work. For completeness, Section 3.2 provides a brief description of the operational information of SPM-SLAM. Section 3.3 describes some improvements that we made.

3.1. Concepts and Notations

Figure 2 shows the relationship between the three reference systems and defines some terms employed in this work. Let us denote f^t as a frame captured by a camera at the time t. The equations of pose transform between the three reference systems are represented by

$$\begin{cases} \boldsymbol{T}_{cw}^{t} = \boldsymbol{T}_{cm}^{t} \cdot (\boldsymbol{T}_{wm})^{-1} \\ \boldsymbol{T}_{wm} = (\boldsymbol{T}_{cw}^{t})^{-1} \cdot \boldsymbol{T}_{cm}^{t} , \\ \boldsymbol{T}_{cm}^{t} = \boldsymbol{T}_{cw}^{t} \cdot \boldsymbol{T}_{wm} \end{cases}$$
(1)

where $T_{cw}^t \in SE(3)$ is the pose that transforms points from the world reference system (*wrs*) to the camera reference system (*crs*) at the time *t*. SE(3) [24] is the group of rigid transformations in 3D space and a 4 × 4 matrix. $T_{wm} \in SE(3)$ is the pose that transforms points from the marker reference system (*mrs*) to the world reference system. $T_{cm}^t \in SE(3)$ is the pose that transforms points from the marker reference system to the camera reference system at the time *t*.



Figure 2. The relationship between three coordinate systems and terms.

We also denote the three-dimensional point $p_i \in \mathbb{R}^3$ (i = 1, 2, 3, 4) as the four corners of marker *m* in the marker reference system, and $u_i^t \in \mathbb{R}^2$ as the pixel locations observed in a frame f^t . Let us denote \tilde{u}_i^t as the projection that can be represented by

$$\widetilde{u}_i^t = \Psi(K, D, T_{cm}^t, p_i), \qquad (2)$$

where *K* and *D* are the camera-intrinsic parameters and distortion coefficients. Naturally, given the transform matrix T_{cm}^t , the reprojection error of a marker's corners in a frame f^t is calculated as:

$$e_m^t(T_{cm}^t) = \sum_{i=1}^4 \|\Psi(K, D, T_{cm}^t, p_i) - u_i^t\|^2.$$
(3)

As mentioned earlier in Section 1, the pose of the marker *m* observed in the frame f^t has two solutions because of marker pose ambiguity. We denote \dot{T}_{cm}^t and \ddot{T}_{cm}^t as the two solutions returned by the IPPE method. The reprojection error *ratio* caused by the poses \dot{T}_{cm}^t and \ddot{T}_{cm}^t is represented by

$$ratio = \frac{\min(e_m^t(\dot{T}_{cm}^t), e_m^t(\ddot{T}_{cm}^t))}{\max(e_m^t(\dot{T}_{cm}^t), e_m^t(\dot{T}_{cm}^t))}.$$
(4)

We denote the angle θ_m^t as the view angle by which the marker *m* is observed in the frame f^t . Considering the translation components t_{cm}^t of the pose T_{cm}^t , we calculate the angle θ_m^t by

$$\theta_m^t = \arccos \frac{z_m^t}{\sqrt{(x_m^t)^2 + (y_m^t)^2 + (z_m^t)^2}}.$$
(5)

where $t_{cm}^t = (x_m^t, y_m^t, z_m^t)$. Since marker pose ambiguity only exists in the rotational components of the pose, the translation components of two solutions (\dot{T}_{cm}^t and \ddot{T}_{cm}^t) are available in (5). Then, the angle θ_{12} between the two frames is represented as:

$$\theta_{12} = \frac{\sum_{i=1}^{n} |\theta_i^{t1} - \theta_i^{t2}|}{n}.$$
(6)

where *i* refers to *n* markers that can be observed in both frames.

r

3.2. SPM-SLAM

In this section, we review some of the SPM-SLAM algorithms, which are essential to the proposed improvements in this paper. As we can observe from Figure 3, SPM-SLAM runs in a single thread. Firstly, it localizes the camera with every frame by finding ArUco tag matches and minimizing the error that derives from reprojecting these tags to the previous frame. Then, it localizes the new tags from a set of ambiguous observations. After that, it uses local bundle adjustment to optimize a set of keyframes and ArUco tags. Finally, once a revisited tag is detected, the system corrects the accumulated drift using pose-graph optimization with the g2o framework [25]. After pose-graph optimization, it performs global optimizes the poses of tags and keyframes.



Figure 3. Pipeline for SPM-SLAM and our system. The green bounding boxes are the methods that we improved.

3.3. Improvements in Our System

This section mainly describes the operational process of the system and the improvements we made. Figure 3 describes the system's operational information. The green bounding boxes are the improvement methods that we proposed. ArUco or AprilTag markers can be detected in our system by the ArUco library [10,11] or AprilTag 3 library [26]. At first, the initialization process establishes the world reference system for the map and adds marker(s) and keyframe(s) to the map. Our improved initialization method (Section 4.1) keeps the process more robust in terms of its rotational movement. After successful initialization, the system enters tracking mode. The tracking process aims to estimate the current frame pose T_{cw}^t by minimizing the reprojection error of the visible markers' corners for each new captured frame f^t [15]. When tracking failure occurs, relocalization is started. Subsequently, the system performs the keyframe decision process. The method of inserting the keyframe and makers to the map is introduced in Section 4.2. The pose(s) T_{wm} of new marker(s) will be estimated later by a new method. This is one of the contributions of this paper. The method estimates the marker pose from multiview frames by minimizing the average rotation between markers (Section 4.3). After the insertion of keyframes and markers, a keyframe culling process is run to delete unnecessary keyframes in the map, which is reduces the processing time of local optimization and global optimization. Local optimization aims to simultaneously optimize the keyframe and marker poses by minimizing the markers' reprojection errors.

The loop closing process is run after the tracking process. It effectively eliminates the errors accumulated along the path [27]. The final step before a new frame is captured is selecting a reference keyframe f^w from the map for the current frame f^t . The process of tracking, loop closure, and local optimization for the next frame f^{t+1} will use the reference keyframe. This is the one that is nearest to the current frame f^t . Finally, the global optimization process is run at the end of the system. All the keyframes and markers of the map are employed in this.

4. Enhanced Methods

4.1. Initialization

The initialization process aims to establish a world reference system for the map, and add keyframes and markers with correct poses to the map, which is vital to an SLAM system. As in SPM-SLAM, we adopt one-frame and two-frame initialization. However, we set different conditions to decide whether the process succeeds.

One-frame initialization succeeds if the *ratio* of at least one marker is lower than the threshold τ_e in SPM-SLAM [15]. However, when the marker is on the edge of an image, it is sometimes misjudged. In our system, the following two conditions must be met for one-frame initialization to succeed.

$$ratio < \tau_e, and \theta_m^r < \tau_\theta, \tag{7}$$

where θ_m^t is calculated by (5). The condition ($\theta_m^t < \tau_{\theta}$) means that a marker *m* must be observed within a proper scope of view. The value τ_{θ} is related to the camera's angle of view (FOV). This is set to 0.4–0.6 times the camera FOV used in our experiment.

For two-frame initialization, our method is the same as that of SPM-SLAM, which estimates the pose of frames and markers that are ambiguously observed in the two frames [15]. Two-frame initialization is successful if one of the following two conditions is met.

$$d(f^0, f^1) > \tau_b, \text{ or } \theta_{12} > \Delta\theta, \tag{8}$$

where $d(f^0, f^1)$ is the distance between f^0 and f^1 . The angle θ_{12} between f^0 and f^1 is calculated by (6). In contrast to SPM-SLAM's method, the condition $\theta_{12} > \Delta\theta$ helps our system to successfully initialize under rotational movements.

4.2. New Keyframes and Markers Insertion

When a frame pose T_{cw}^t is successfully estimated in the tracking process, we need to decide whether the frame can be judged as a keyframe and inserted into the map. The insertion of keyframes has a heavy influence on local optimization and tracking. Too many keyframes may cause local optimization to consume a high amount of computational resources, while fewer keyframes may lead to a poor tracking performance, or even tracking

failure. Therefore, keyframes can be inserted into the map when one of the following conditions are satisfied:

- 1. If at least one new marker is detected in a frame, the frame is spawned as a new keyframe and added to the map with the new marker(s). Notably, the pose(s) of the new marker(s) are set to zero. Later, these are estimated from multiple views (Section 4.3).
- 2. If the distance between the current frame f^t and the reference keyframe f^w is larger than the threshold τ_b , the frame is added to the map.
- 3. If the angle between the current frame f^t and the reference keyframe f^w is larger than the threshold $\Delta \theta$, the frame is added to the map.

Unlike SPM-SLAM, which only uses conditions 1 and 2, the use of condition 3 ensures that our system instantly inserts new keyframes during rotational movements. This improves the robustness of the system, allowing for it to adapt to more environments.

4.3. Marker Pose Estimation

Once a new marker *m* is added to the map, the marker poses T_{wm} should rapidly be obtained. We propose a novel method to estimate the marker pose from a set of keyframes that observe that marker *m*. Let us denote \mathcal{F} as the set of keyframes. The poses of \mathcal{F} with keyframes are represented by $(T_{cw}^1, T_{cw}^2, \dots, T_{cw}^n)$, which are known after the tracking process. Considering the pose ambiguity of marker *m* in the set of keyframes \mathcal{F} , we can obtain possible poses for the marker *m* according to (1), which is denoted by:

$$\Phi(m) = \{ \dot{\boldsymbol{T}}_{wm}^{1}, \ddot{\boldsymbol{T}}_{wm}^{1}, \cdots, \dot{\boldsymbol{T}}_{wm}^{n}, \ddot{\boldsymbol{T}}_{wm}^{n} \}$$
(9)

Our goal is to choose the best pose T_{wm} of the marker *m* from $\Phi(m)$. Given a set of \mathcal{F} , the relative rotations from marker *i* to marker *j* (M2M) have four solutions, which are represented by

$$\begin{cases} \mathbf{R}_{i,j}^{t,00} = (\dot{\mathbf{R}}_{j}^{t})^{T} \cdot \dot{\mathbf{R}}_{i}^{t} \\ \mathbf{R}_{i,j}^{t,01} = (\ddot{\mathbf{R}}_{j}^{t})^{T} \cdot \dot{\mathbf{R}}_{i}^{t} \\ \mathbf{R}_{i,j}^{t,10} = (\dot{\mathbf{R}}_{j}^{t})^{T} \cdot \ddot{\mathbf{R}}_{i}^{t} \\ \mathbf{R}_{i,j}^{t,11} = (\ddot{\mathbf{R}}_{j}^{t})^{T} \cdot \ddot{\mathbf{R}}_{i}^{t} \end{cases}$$
(10)

where $\{\dot{\mathbf{R}}_{i}^{t}, \ddot{\mathbf{R}}_{i}^{t}\}$ are the rotational components of the two ambiguous poses of marker *i* in the frame f^{t} . They have a 3 × 3 matrix. Let us denote $\{M_{i}\}_{i=1}^{p}$ as the observed markers with the correct poses from the set of keyframes \mathcal{F} . Remember that these poses are known, and that we denote $\{R_{w1}, \dots, R_{wp}\}$ as the rotational components.

Our method is to select the best pose T_{wm} from $\Phi(m)$ by minimizing the M2M error between marker *m* and markers $\{M_i\}_{i=1}^p$ in the set of keyframes \mathcal{F} . The M2M error is represented by

$$e(\mathbf{T}_{wm}) = \sum_{t=1}^{n} \sum_{i=1}^{p} \min \begin{pmatrix} \| (\mathbf{R}_{wi})^{T} \cdot \mathbf{R}_{wm} - \mathbf{R}_{m,i}^{t,00} \|_{F}^{2} \\ \| (\mathbf{R}_{wi})^{T} \cdot \mathbf{R}_{wm} - \mathbf{R}_{m,i}^{t,01} \|_{F}^{2} \\ \| (\mathbf{R}_{wi})^{T} \cdot \mathbf{R}_{wm} - \mathbf{R}_{m,i}^{t,10} \|_{F}^{2} \\ \| (\mathbf{R}_{wi})^{T} \cdot \mathbf{R}_{wm} - \mathbf{R}_{m,i}^{t,11} \|_{F}^{2} \end{pmatrix}$$
(11)

where R_{wm} is the rotational components of T_{wm} , and $\|\cdot\|_F^2$ denotes the Frobenius norm. The best solution is the one that minimizes the M2M error $e(T_{wm})$ in the set of keyframes \mathcal{F} . Note that the translation component is already included in T_{wm} . Therefore, we can find a correct pose for marker m by (11). Furthermore, the angle or distance between the keyframes in \mathcal{F} must be larger than threshold $\Delta\theta$ or threshold τ_b , respectively. This could make (11) more effective. Note that the parameters $\Delta\theta$ and τ_b are the same as those mentioned in Section 4.2.

4.4. Relocalization

The relocalization is run when the system fails to track. Let us denote \mathcal{M} as a marker with the correct pose T_{wm} detected in current frame f^t , and n is the number of markers in \mathcal{M} .

If n < 1, the process will fail.

If n = 1, two methods can be used to obtain a solution. One is that, if the marker satisfies condition (7), we can obtain the marker's unambiguous pose T_{cm}^t . Then, we can successfully compute the current frame pose T_{cw}^t using (1). The other method is that, if the marker is not satisfied with the condition (7), we can obtain the marker's ambiguous poses \dot{T}_{cm}^t and \ddot{T}_{cm}^t using (1). Naturally, the two poses (\dot{T}_{cw}^t and \ddot{T}_{cw}^t) of the current frame are computed. Then, the two distances between the current frame and the reference frame are calculated using the two poses. Finally, the pose with the smaller distance is the correct one.

If n > 1, let us denote

$$\Omega(T_{cn}^{t}) = \{ \dot{T}_{c1}^{t}, \ddot{T}_{c1}^{t}, \cdots, \dot{T}_{cn}^{t}, \ddot{T}_{cn}^{t} \},$$
(12)

as the ambiguous poses of markers \mathcal{M} in current frame f^t , and denote

$$\mathbf{Y}(\boldsymbol{T}_{wn}) = \{\boldsymbol{T}_{w1}, \cdots, \boldsymbol{T}_{wn}\},\tag{13}$$

as the correct poses. The M2M error in \mathcal{M} is represented by:

$$e < i, j >= \| (\mathbf{R}_{ci})^T \cdot \mathbf{R}_{cj} - (\mathbf{R}_{wi})^T \cdot \mathbf{R}_{wj} \|_F^2,$$
(14)

where R_{ci} (3 × 3 matrix) is the rotational component of $T_{cn}^t \in \Omega(T_{cn}^t)$, and R_{wi} is the rotational component of $T_{wn} \in Y(T_{wn})$. Here, note that if the right rotational component is selected, the translation component is also selected. Accordingly, our goal is firstly to select a pair of poses T_{ci}^t and T_{cj}^t from $\Omega(T_{cn}^t)$ by minimizing the M2M error (14), and then calculate the pose with (1). As the correct poses ({ T_{wi}, T_{wj} } $\in Y(T_{wn})$) are known, we can obtain two solutions ($T_{cw_i}^t$ and $T_{cw_j}^t$) for the current frame pose. If the distance between $T_{cw_i}^t$ and $T_{cw_j}^t$ is lower than the threshold σ_e (default is 0.01m in our system), the relocalization process will succeed and the two poses ($T_{cw_i}^t$ and $T_{cw_j}^t$) are available. Otherwise, relocalization will fail.

5. Experiments and Results

This section evaluates our method's performance in the publicly available SPM dataset [15] and our dataset. The SPM dataset has eight sequences and provides ground truth trajectories for each sequence. Our dataset contains six indoor sequences and two outdoor sequences, which include more rotational movements. We aim to evaluate the tracking accuracy and robustness of our methods. Since the SPM dataset offers the ground truth, it was used to evaluate the tracking accuracy. We assessed the robustness of our system on our dataset, which includes many rotational movements. All tests were run on an Intel Core i5-11600K 3.9 GHz desktop computer.

In addition, our method can be compared with other marker-based SLAM systems, which include SPM-SLAM, TagSLAM, and UcoSLAM, as SPM-SLAM, UcoSLAM, and our system have the same parameters. In order to be comparable, these same parameters must be equal. The parameters τ_b and τ_e (Section 4) employed in all three systems were set at 7 mm and 0.333 in the two datasets. The other parameters $\Delta\theta$ and σ_e , which were only employed in our system, were set at 5° and 0.01 m, which proved to be good values in most of our experiments.

5.1. SPM Dataset

The SPM dataset contains eight video sequences, with fiducial markers recorded in a laboratory, and provides a ground truth. The camera works at 60 Hz with a resolution after rectification of 1920×1080 pixels. The sequences 01, 02, 03, 07, and 08 were recorded by a camera pointed toward the walls, and others were recorded by a camera pointed towards the ceiling. We evaluated the accuracy of the trajectory using the Absolute Trajectory Error (ATE) measure, which is the translational RMSE. In addition, the system's mean speed was also considered as an evaluation indicator.

SPM-SLAM, TagSLAM, and UcoSLAM were selected for comparison in the experiments. Table 1 shows the ATEs and Trck of the methods tested on the eight sequences. Figure 4 shows the trajectories for ground truth (dash gray), SPM-SLAM (blue), TagSLAM (green), UcoSLAM (red), and ours (purple). As can be observed, TagSLAM's accuracy was the worst, especially in sequence 01, where its ATE was 0.43 m. Compared with SPM-SLAM and UcoSLAM, our method's accuracy was slightly better in most sequences. Figure 5 shows the mean speed of the four systems. The graph shows that TagSLAM and UcoSLAM are much slower than other methods. This is because TagSLAM considers every frame to be a keyframe and UcoSLAM combines keypoints and markers, forming the CPU load. In contrast, our method and SPM-SLAM can run at about 145 frames per second (fps) on the SPM dataset.

Table 1. Absolute trajectory errors on the SPM dataset.

Dataset	Length [m]	SPM-SLAM	TagSLAM	UcoSLAM	Ours
		ATE [m]			
sequence 01	19.5	0.060	0.430	0.084	0.059
sequence 02	23.3	0.046	0.054	0.079	0.045
sequence 03	23.1	0.055	0.233	0.058	0.054
sequence 04	32.8	0.014	0.027	0.037	0.013
sequence 05	26.9	0.017	0.023	0.070	0.016
sequence 06	33.2	0.017	0.026	0.028	0.016
sequence 07	18.9	0.048	0.246	0.094	0.050
sequence 08	26.1	0.064	0.066	0.131	0.063



Figure 4. SLAM trajectories in SPM dataset. The dashed gray line is the ground truth. The blue line is SPM-SLAM. The green line is TagSLAM. The red line is UcoSLAM. The purple line is ours.



Figure 5. The mean speed of four systems (higher is better).

5.2. Our Dataset

160

120

80

40

0

Mean speed [frames/s]

The test on the public dataset showed that our method significantly outperforms TagSLAM and UcoSLAM in accuracy and speed. However, the accuracy of our method is slightly better than SPM-SLAM in most sequences, and the speed is almost the same as SPM-SLAM. Therefore, to further evaluate that our method is more robust than SPM-SLAM in terms of rotational movements, it was tested and compared with SPM-SLAM on our dataset.

For this test, we recorded eight video sequences. Sequences 01-06 were recorded in a laboratory, and 22 markers were placed on the wall in an approximated dimension of 1.5×1.5 square meters. Sequences 07 and 08 were recorded outdoors, and 48 markers were placed on the ground in an approximated dimension of 20×15 and 30×20 square meters. Figure 6 shows the markers on the ground for sequences 07 and 08. It is worth noting that outdoor sequences 07 and 08 were recorded with a camera mounted on a vehicle. The camera captured images at a 1920×1080 pixel resolution and a frame rate of 30Hz. For sequences 01 and 02, the camera was pointed towards the wall and was spinning. Furthermore, sequence 01 was rotated about 270 degrees, and sequence 02 started and ended at the same position to facilitate loop-closure detection. Different from sequences 01– 02, sequences 03–06 have more positional movements and only have rotational movements at the corners. For outdoor sequences, sequence 08 showed more rotational movement than sequence 07. Sequences 02–08 can facilitate the detection of loop closure. Although we could not obtain the ground truth for the camera poses, the tracking success rate (Trck) and the number of markers reconstructed in the mapping were used to evaluate the system's robustness. The Trck is the ratio of the number of tracked frames to the total number of frames in a sequence.



(a) sequence 07

(b) sequence 08

Figure 6. (a,b) show the markers on the ground for outdoor sequences 07 and 08.

Table 2 shows the number reconstructed markers and the tracking rate. As can be observed, the SPM-SLAM Trcks were lower than 55% on sequences 01–03 and 05–06, while the proposed method achieved over 99% on all sequences. Additionally, SPM-SLAM only reconstructed all markers in sequence 07. In contrast, our method reconstructed all markers in all sequences. Figure 7 shows the reconstruction results of both methods for sequences 01 and 02. This clearly shows that more markers were reconstructed by our method than SPM-SLAM. Figure 8a–h show the reconstructed trajectory of both methods for sequences 03–06. We can observe that our method outperforms SPM-SLAM, especially in sequences 03

and 06, where SPM-SLAM almost completely failed to reconstruct the trajectory. Figure 8i–l show the reconstructed trajectory of both methods for sequences 07 and 08 (outdoors). As can be observed, our method was slightly better than SPM-SLAM on sequence 07 and much better than SPM-SLAM on sequence 08. This is not unexpected, since sequence 08 has more rotational movement than sequence 07. In summary, the experiments show that the robustness of our system is better than that of the SPM-SLAM under rotational movements.



Figure 7. (a,b) mapping results of SPM-SLAM on sequences 01 and 02. (c,d) results for our dataset.



Figure 8. Trajectories on sequences 03-08 in our dataset. (**a**–**d**,**i**,**k**) are generated by SPM-SLAM. (**e**–**h**,**j**,**l**) are generated by ours.

Datacat	Total Number of Markers	SPM-SLAM	Ours	
Dataset	Total Number of Warkers —	Num of Reconstructed Markers/Trck		
sequence 01	12	8/72.6%	12 /99.8%	
sequence 02	22	6/13.4%	22/99.9%	
sequence 03	22	8/16.1%	22/99.9%	
sequence 04	22	20/86.2%	22/99.5%	
sequence 05	22	11/54.7%	22/99.9%	
sequence 06	22	2/5.2%	22/99.9%	
sequence 07	48	48/99.4%	48/99.9%	
sequence 08	48	47/93.7%	48/99.2%	

Table 2. The number of markers reconstructed in the mapping results and tracking success rate of frames.

6. Conclusions

This paper presents an improved version of the existing SPM-SLAM. To improve the robustness and accuracy, we improved the initialization, keyframe insertion and relocalization algorithms based on the SPM-SLAM dataset and proposed a method to solve the problem of planar-marker pose ambiguity. Our system was compared with the state-of-theart, such as SPM-SLAM, TagSLAM, and UcoSLAM. The results on public datasets show that our system achieves better accuracy in most sequences, and the speed of our system is much faster than that of UcoSLAM and TagSLAM. The results of our datasets show that our method is more robust than SPM-SLAM under rotational movements and reconstructs more planar markers in the map. Additionally, our system runs at approximately 145 Hz using a single thread, which benefits UAVs and mobile robots, as they require fewer computational resources. It is worth mentioning that the open-source code is available on GitHub (https://github.com/BIT-wangzb/Marker-based-SLAM) (accessed on 10 December 2022).

Author Contributions: Conceptualization, Z.W. and Z.Z.; methodology, Z.W. and W.Z.; investigation, Z.W. and X.H.; software, Z.W.; validation, Z.W. and H.D.; visualization, Z.W.; writing—original draft, Z.W.; supervision, G.H., X.K. and W.Z.; funding acquisition, Z.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 61773059, and the National Defense Technology of China, grant number 2020202020028.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

- Qin, T.; Li, P.; Shen, S. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. *IEEE Trans. Robot.* 2018, 34, 1004–1020. [CrossRef]
- Campos, C.; Elvira, R.; Rodríguez, J.J.G.; Montiel, J.M.M.; Tardós, J.D. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial and Multi-Map SLAM. *IEEE Trans. Robot.* 2021, 37, 1874–1890. Available online: https://ieeexplore.ieee.org/ abstract/document/9440682 (accessed on 10 December 2022). [CrossRef]
- Ortiz-Fernandez, L.E.; Cabrera-Avila, E.V.; da Silva, B.M.F.; Gonçalves, L.M.G. Smart Artificial Markers for Accurate Visual Mapping and Localization. Sensors 2021, 21, 625. [CrossRef] [PubMed]
- 4. Mur-Artal, R.; Tardos, J.D. Visual-Inertial Monocular SLAM With Map Reuse. *IEEE Robot. Autom. Lett.* 2017, 2, 796–803. [CrossRef]
- 5. Daniilidis, K. Hand-eye calibration using dual quaternions. Int. J. Robot. Res. 1999, 18, 286–298. [CrossRef]
- Mur-Artal, R.; Tardós, J.D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* 2017, 33, 1255–1262. [CrossRef]
- Engel, J.; Schöps, T.; Cremers, D. LSD-SLAM: Large-scale direct monocular SLAM. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; Springer: Berlin/Heidelberg, Germany, 2014; pp. 834–849.

- Gao, X.; Wang, R.; Demmel, N.; Cremers, D. LDSO: Direct sparse odometry with loop closure. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 2198–2204.
- 9. Huang, B.; Zhao, J.; Liu, J. A survey of simultaneous localization and mapping with an envision in 6g wireless networks. *arXiv* **2019**, arXiv:1909.05214.
- 10. Garrido-Jurado, S.; Muñoz-Salinas, R.; Madrid-Cuevas, F.J.; Marín-Jiménez, M.J. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognit.* 2014, 47, 2280–2292. [CrossRef]
- 11. Garrido-Jurado, S.; Munoz-Salinas, R.; Madrid-Cuevas, F.J.; Medina-Carnicer, R. Generation of fiducial marker dictionaries using mixed integer linear programming. *Pattern Recognit.* **2016**, *51*, 481–491. [CrossRef]
- 12. Olson, E. AprilTag: A robust and flexible visual fiducial system. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Qingdao, China, 17–19 September 2011; pp. 3400–3407.
- Kato, H.; Billinghurst, M. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99), Washington, DC, USA, 20–21 October 1999; pp. 85–94.
- 14. Muñoz-Salinas, R.; Marín-Jimenez, M.J.; Yeguas-Bolivar, E.; Medina-Carnicer, R. Mapping and localization from planar markers. *Pattern Recognit.* **2018**, *73*, 158–171. [CrossRef]
- 15. Munoz-Salinas, R.; Marin-Jimenez, M.J.; Medina-Carnicer, R. SPM-SLAM: Simultaneous localization and mapping with squared planar markers. *Pattern Recognit.* 2019, *86*, 156–171. [CrossRef]
- 16. Munoz-Salinas, R.; Medina-Carnicer, R. UcoSLAM: Simultaneous localization and mapping by fusion of keypoints and squared planar markers. *Pattern Recognit.* 2020, 101, 107193. [CrossRef]
- 17. Pfrommer, B.; Daniilidis, K. Tagslam: Robust slam with fiducial markers. arXiv 2019, arXiv:1910.00679.
- Oberkampf, D.; DeMenthon, D.F.; Davis, L.S. Iterative pose estimation using coplanar feature points. *Comput. Vis. Image Underst.* 1996, 63, 495–511. [CrossRef]
- Schweighofer, G.; Pinz, A. Robust pose estimation from a planar target. *IEEE Trans. Pattern Anal. Mach. Intell.* 2006, 28, 2024–2030. [CrossRef] [PubMed]
- 20. Collins, T.; Bartoli, A. Infinitesimal plane-based pose estimation. Int. J. Comput. Vis. 2014, 109, 252–286. [CrossRef]
- 21. Kuutti, S.; Fallah, S.; Katsaros, K.; Dianati, M.; Mccullough, F.; Mouzakitis, A. A survey of the state-of-the-art localization techniques and their potentials for autonomous vehicle applications. *IEEE Internet Things J.* **2018**, *5*, 829–846. [CrossRef]
- Kaess, M.; Johannsson, H.; Roberts, R.; Ila, V.; Leonard, J.; Dellaert, F. iSAM2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 3281–3288.
- Ch'ng, S.F.; Sogi, N.; Purkait, P.; Chin, T.J.; Fukui, K. Resolving marker pose ambiguity by robust rotation averaging with clique constraints. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, May 31–August 31 2020; pp. 9680–9686.
- 24. Eade, E. Lie Groups for 2d and 3d Transformations. 2013, Volume 117, p. 118. Available online: http://ethaneade.com/lie.pdf (accessed on 10 December 2022).
- Kümmerle, R.; Grisetti, G.; Strasdat, H.; Konolige, K.; Burgard, W. g 2 o: A general framework for graph optimization. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 3607–3613.
- Krogius, M.; Haggenmiller, A.; Olson, E. Flexible layouts for fiducial tags. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 4–8 November 2019; pp. 1898–1903.
- Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* 2015, 31, 1147–1163. [CrossRef]

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.