2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

MCGMapper: Light-Weight Incremental Structure from Motion and Visual Localization With Planar Markers and Camera Groups

Yusen Xie, Zhenmin Huang, Kai Chen, Lei Zhu, and Jun Ma

Abstract— **Structure from Motion (SfM) and visual localization in indoor texture-less scenes and industrial scenarios present prevalent yet challenging research topics. Existing SfM methods designed for natural scenes typically yield low accuracy or map-building failures due to insufficient robust feature extraction in such settings. Visual markers, with their artificially designed features, can effectively address these issues. Nonetheless, existing marker-assisted SfM methods encounter problems like slow running speed and difficulties in convergence; and also, they are governed by the strong assumption of unique marker size. In this paper, we propose a novel SfM framework that utilizes planar markers and multiple cameras with known extrinsics to capture the surrounding environment and reconstruct the marker map. In our algorithm, the initial poses of markers and cameras are calculated with Perspective-n-Points (PnP) in the front-end, while bundle adjustment methods customized for markers and camera groups are designed in the back-end to optimize the 6-DOF pose directly. Our algorithm facilitates the reconstruction of large scenes with different marker sizes, and its accuracy and speed of map building are shown to surpass existing methods. Our approach is suitable for a wide range of scenarios, including laboratories, basements, warehouses, and other industrial settings. Furthermore, we incorporate representative scenarios into simulations and also supply our datasets with pose labels to address the scarcity of quantitative ground-truth datasets in this research field. The datasets and source code are available on GitHub[1].**

I. INTRODUCTION

Structure from Motion (SfM) and visual localization are significant research topics in the context of computer vision. Numerous cutting-edge methods in these areas depend on the extraction of human-defined feature points from images of environments [1]–[4]. Those feature points are employed to establish associations across different frames such that the principle of stereo vision can be applied to recover 3D structure. Nevertheless, these approaches generally struggle in scenarios that are either texture-less or characterized by repetitive patterns (such as corridors, warehouses, etc.), where extractable feature points may be insufficient or erroneous associations between feature points from different images may be established. In such cases, man-made visual markers, known for their distinct and identifiable attributes, can be effectively deployed to resolve these issues by providing accurate and stable visual constraints. Consequently, efforts have been directed towards developing algorithms that capitalize on the advantages of visual markers, leading to precise and stable reconstruction and visual localization in these challenging scenarios [5]–[11].

In marker-assisted SfM algorithms, visual markers are typically used to provide supplementary information to establish stable and reliable correspondences between feature points extracted from different images. Subsequently, multi-view triangulation is performed to estimate the spatial locations of these feature points, followed by a nonlinear optimization process to further refine the locations and recover the camera poses. However, these methods exhibit certain limitations. Firstly, although markers of different sizes can definitely provide more hierarchical and detailed representations of the scenes, existing methods in the literature are only applicable to markers of a unique size. Secondly, the optimization process in existing methods generally suffers from difficulties in convergence, resulting in slow convergence rates, local optima, or even failures. Thirdly, they exclusively support the use of a single monocular camera rather than camera groups (CG), and thus the resulting insufficient field of view could lead to failures such as discontinuous map reconstruction [5], [8] and ambiguity in the marker pose [11].

To address all these issues, we propose MCGMapper, an incremental marker-CG SfM framework for marker map reconstruction and visual localization. More specifically, our contributions are as follows:

1) We derive novel bundle adjustment for markers of different sizes and indefinite number of cameras in camera groups, where the orthogonality constraints of the marker's corner points and the extrinsic between the cameras in camera groups are considered as prior knowledge, facilitating the consideration of the inherent geometrical constraints of markers and camera groups.
2) We propose an incremental SfM framework that integrates the front-end PnP with our proposed customised bundle adjustment in the back-end, achieving high accuracy and rapid convergence speed of mapping.
3) We introduce an optimization-based localization algorithm that utilizes a global marker map as a reference, and all observations from a frame are introduced to av-

This work was supported in part by the National Natural Science Foundation of China under Grant 62303390; in part by the Guangzhou-HKUST(GZ) Joint Funding Scheme under Grants 2023A03J0148 and 2024A03J0618; and in part by the Project of Hetao Shenzhen-Hong Kong Science and Technology Innovation Cooperation Zone under Grant HZQB-KCZYB-2020083. *(Corresponding Author: Jun Ma.)*

Yusen Xie, Zhenmin Huang, Kai Chen, and Lei Zhu are with the Robotics and Autonomous Systems Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China (e-mail: yxie827@connect.hkust-gz.edu.cn; zhuangdf@connect.ust.hk; kchen916@connect.hkust-gz.edu.cn; leizhu@ust.hk).

Jun Ma is with the Robotics and Autonomous Systems Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China, also with the Division of Emerging Interdisciplinary Areas, The Hong Kong University of Science and Technology, Hong Kong SAR, China, and also with HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute, Futian, Shenzhen, China (e-mail: jun.ma@ust.hk).

[1]https://github.com/xieyuser/MCGMapper

erage the error, resulting in accurate global localization.

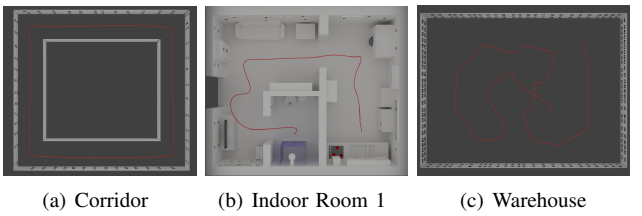
4) We contribute a synthetic marker dataset comprising markers of various sizes and ground truth pose labels to facilitate the quantitative analysis in the field of marker-based reconstruction.

II. Related works

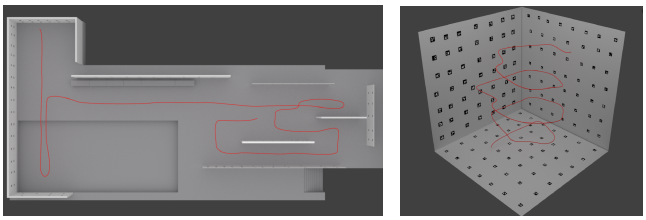
SfM is a long-existing research area in computer vision, which aims to recover 3D structures of target objects from a set of multi-view 2D images through the principle of stereo vision. In SfM, a key step is to perform stereo matching between images. Remarkably, in many existing methods, this step mainly relies on extracting human-defined feature points from images and compare their descriptors. For example, in [2]–[4], [12], SIFT points [13] are extracted from images and matched into point pairs, which are then processed by the 5-point [14] or 8-point method [15] to obtain the fundamental matrices. After feature matching, multi-view triangulation is performed to estimate the spatial location of those feature points, followed by a bundle adjustment to further refine the locations and obtain an accurate 3D map. These methods manage to reconstruct Rome's landmarks from unordered collection of images. However, these methods generally exhibit poor performance in dimly lit indoor scenes or industrial scenes with repetitive patterns, as stable visual features are sparse, or there are easily suggested erroneous correspondences between them.

With its stable artificial features, visual marker can be utilized to mitigate the problem and boost up the performance of SfM. Within the realm of SfM methods utilizing visual markers, a portion of them are not dependent on visual feature points. Examples include MarkerMapper [5], [6], where only visual markers are used to provide correspondence. In these methods, a graph connection structure between poses is constructed in the front-end, and graph connection path is searched to minimize the reprojection error, followed by a global bundle adjustment for refinement. Due to its strong dependence on the initial calculation of the path, this method often fails in large scenes. On the contrary, Degol [7] utilizes both marker and feature points, where markers are used to provide information for robust image correspondence. Compared to MarkerMapper, it exhibits stable reconstruction without discontinuity in the resulting map. However, this method is time-consuming due to plenty of feature points involved. PytagMapper [8] uses Gaussian Belief Propagation [16] to perform optimization in the back-end, but this method encounters challenges in achieving convergence. A graph filtering method [11] is proposed to guide the feature matching process with the absence of quantitative analysis.

Meanwhile, all the marker reconstruction algorithms mentioned above rely on a single monocular camera, which can easily lead to discontinuous maps [5] and ambiguities in the marker poses [9] due to small FOV and limited information perceived in each image. With the development of fusion algorithm in camera groups, MultiCol [17] develops a hypergraph formulation of multiple cameras, yet it frequently experiences tracking failures in low-illumination environments.



(a) Corridor (b) Indoor Room 1 (c) Warehouse



(d) Swimming Pool (e) Calibration Room

Fig. 1. The scenes simulated using Blender [20]. The red curve represents the movement trajectory of the camera or camera groups in each scene.

Eckenhoff [18] proposes the minimize preintegration pose error term based on spatial and temporal relations. Marcus [19] implements multiple cameras for visual odometry in indoor parking lots, resulting in effective performance for automated parking. Although better results are achieved in specific scenarios, the above methods still treat each camera in a camera groups separately and fail to consider the given transformation constraints between cameras.

In summary, existing marker-based SfM approaches encounter difficulties in reconstructing extensive, texture-less areas, and they cannot effectively handle different marker sizes and camera groups. This greatly limits the application of artificial markers in industrial scenes. Additionaly, we also recognize that there is a scarcity of available datasets for marker assisted SfM. Although a small 7×7 meter-sized room from MarkerMapper [5], [6] and a collection of 16 unordered images from Degol [7] are provided, ground-truth labels that are essential for quantitative evaluation are not contained.

III. Methodology

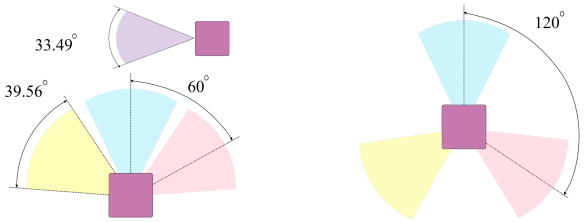
A. Synthetic Datasets with Ground Truth Labels

Currently, marker-assisted visual SfM lacks datasets with ground-truth pose labels. To facilitate quantitative evaluation of algorithms, we build several synthetic scenes in the 3D simulation software Blender [20], including indoor rooms, corridors, warehouses, etc., with dimensions ranging from meters to tens of meters. Fig. 1 illustrates the synthetic scenes and camera trajectory.

TABLE I

The detailed information of each scenario

Sequence	Camera Setups	Markers	Images	Dimensions (Meter)
Indoor Room 1	Mono/CG-60/CG-120	60 (same)	65×8	9×7×2.5
Indoor Room 2	Mono/CG-60/CG-120	100 (same)	92×8	11.4×8.2×4
Corridor	Mono/CG-60/CG-120	200 (different)	90×8	23×21×2
Warehouse	Mono/CG-60/CG-120	200 (different)	129×8	20.6×15.2×2
Swimming Pool	Mono/CG-60/CG-120	200 (same)	201×8	53.7×22.8×5
Calibration Room	Mono/CG-60/CG-120	200 (different)	169×3	10×10×10



(a) 60° setup (CG-60) (b) 120° setup (CG-120)

Fig. 2. Camera groups setup in our simulated datasets. FoV of the camera (39.56°× 33.49°) is calculated with the calibrated camera intrinsic. In this paper, we use three cameras to evaluate our algorithm.

Based on these synthetic scenes, we create a new dataset encompassing several scenarios. Table I provides detailed information on each dataset, including the number of markers in each environment (with 'same' and 'different' representing markers of the same or different sizes), the collection positions in the trajectory, the number of images captured at each position, and the dimensions of the scenes. For each dataset, we use three different camera setups to capture the environment, including monocular, CG-60 in Fig. 2(a), and CG-120 in Fig. 2(b). The encoding pattern of markers used in the proposed dataset is Aruco_4×4_1000 [21]. Image resolutions are all 1224×1024 pixels. In contrast with existing datasets that only contain image collections or continuous videos, our dataset provides ground-truth pose labels, enabling rigorous quantitative evaluation of algorithm performance. The ground-truth pose labels are output in TUM [22] format, providing a valuable resource for future research in marker-based SfM.

B. Notations

In this work, $(\cdot)_w$ denotes the world coordinate. $P_w = \{X^1, X^2, ..., X^i, ..., X^I\}$ denotes the object points in the world coordinate $(\cdot)_w$, where I is the number of object points. $F = \{F^1, F^2, ..., F^j, ..., F^J\}$ denotes captured frames, where J is the number of frames. $(\cdot)_{Fj}$ denotes the j-th frame coordinate. Correspondingly, the coordinate of object point i in frame F^j is defined as $X^i_{Fj} = [x^i_{Fj}, y^i_{Fj}, z^i_{Fj}]^\top$. Markers in the scenerio are defined as $M = \{m^1, m^2, ..., m^l, ..., m^L\}$, where L is the number of markers. We also define the set of indices for object points captured by frame F^j as $\mathbb{C}^j$, and further, the measurement of image coordinate for the i-th object point in the j-th frame is defined as x^{ij} for $i \in \mathbb{C}^j$ and $j \in \{0, 1, ..., J\}$. We define the notation $\hat{(\cdot)}$ as $\delta\hat{\xi} = \begin{bmatrix} [\varphi]_\times & \rho \\ 0^\top & 0 \end{bmatrix} \in \mathbb{R}^{4\times4}$, where $\delta\xi = \begin{bmatrix} \rho \\ \varphi \end{bmatrix} \in \mathbb{R}^6$, ρ is the translation vector and φ is the rotation vector, $\rho \in \mathbb{R}^3, \varphi \in \mathbb{R}^3$. Notation $[\cdot]_\times$ represents the skew symmetric matrix corresponding to this vector.

C. Marker Bundle Adjustment

Artificial markers have orthogonal constraints among their corner points, which means that using general bundle adjustment is hard to guarantee the preservation of this constraint. Hence, in this part, we introduce corner points as prior information to design marker bundle adjustment.

In the following part, notations for markers are defined: $(\cdot)_{m^l}$ denotes marker coordinate system, the corner points of the l-th marker are defined as $P^{l^n}_{m^l} = \left\{X^{l^1}_{m^l}, X^{l^2}_{m^l}, X^{l^3}_{m^l}, X^{l^4}_{m^l}\right\} = \left[\left[-s^l, s^l, 0, 1\right], \left[s^l, s^l, 0, 1\right], \left[s^l, -s^l, 0, 1\right], \left[-s^l, -s^l, 0, 1\right]\right]$, where $n \in \{1, 2, 3, 4\}$ and s^l represents the half side length of l-th marker. The transformation from l-th marker coordinate system $(\cdot)_{m^l}$ to the world coordinate $(\cdot)_w$ is denoted by $\mathbf{T}^w_{m^l}$.

Notice that the set of markers captured by frame F_j is defined as $\mathbb{C}^{jm}$, and $l \in \mathbb{C}^{jm}$ in F_j. $[x^{l^n}_{Fj}, y^{l^n}_{Fj}, z^{l^n}_{Fj}]^\top$ are the corresponding points of $P^{l^n}_{m^l}$ in $(\cdot)_{Fj}$. The four corner points in image coordinate projected are defined as $p^{l^n}_{m^l} = \left\{x^{l^1}_{m^l}, x^{l^2}_{m^l}, x^{l^3}_{m^l}, x^{l^4}_{m^l}\right\}$. The projection procedure from $X^{l^n}_{m^l}$ to $x^{l^n}_{m^l}$ is $x^{l^n}_{Fj} = \pi_{\mathbf{K}}(\mathbf{T}^{Fj}_w [\mathbf{T}^{m^l}_w]^{-1} X^{l^n}_{m^l})$. By adding Lie perturbation variables $\delta\xi^m$ to the marker pose $\mathbf{T}^{m^l}_w$, projection error χ^{jmn} can be defined as

$$\begin{aligned} \chi^{jmn} &= x^{l^n}_{m^l} - \pi_{\mathbf{K}}(X^{l^n}_{Fj}) \\ X^{l^n}_{Fj} &= \mathbf{T}^{Fj}_w [\exp(\delta\hat{\xi}^m)\mathbf{T}^{m^l}_w]^{-1} X^{l^n}_{m^l} \end{aligned} \tag{1}$$

Subsequently, we can obtain the partial derivatives of $P^{l^n}_{Fj}$ with respect to the marker pose increment $\delta\xi^m$, which is given as

$$\begin{aligned} \frac{\partial X^{l^n}_{Fj}}{\partial \delta\xi^m} &= \lim_{\delta\xi^m \to 0} \frac{\mathbf{T}^{Fj}_w [\exp(\delta\hat{\xi}^m)\mathbf{T}^{m^l}_w]^{-1} X^{l^n}_{m^l} - \mathbf{T}^{Fj}_w [\mathbf{T}^{m^l}_w]^{-1} X^{l^n}_{m^l}}{\delta\xi^m} \\ &= \lim_{\delta\xi^m \to 0} \frac{\mathbf{T}^{Fj}_w [\mathbf{T}^{m^l}_w]^{-1} \exp(\delta\hat{\xi}^m)^{-1} X^{l^n}_{m^l}}{\delta\xi^m} \\ &= \lim_{\delta\xi^m \to 0} \frac{\mathbf{T}^{Fj}_w \mathbf{T}^w_{m^l} \exp(-\delta\hat{\xi}^m) X^{l^n}_{m^l}}{\delta\xi^m} = \lim_{\delta\xi^m \to 0} \frac{\mathbf{T}^{Fj}_{m^l} (\mathbf{I} - \delta\hat{\xi}^m) X^n_{m^l}}{\delta\xi^m} \\ &= \mathbf{T}^{Fj}_{m^l} \times \begin{bmatrix} [X^{l^n}_{m^l}]_\times & -\mathbf{I}_3 \\ 0 & 0 \end{bmatrix} = \mathbf{R}^{Fj}_{m^l} \cdot \begin{bmatrix} [X^{l^n}_{m^l}]_\times & -\mathbf{I}_3 \end{bmatrix}_{3\times6} \end{aligned} \tag{2}$$

Finally, the partial derivatives of χ^{jmn} with respect to the marker pose increment $\delta\xi^m$ is

$$\begin{aligned} \frac{\partial \chi^{jmn}}{\partial \delta\xi^m} &= \frac{\partial \chi^{jmn}}{\partial P^{l^n}_{Fj}} \cdot \frac{\partial P^{l^n}_{Fj}}{\partial \delta\xi^m} \\ &= \begin{pmatrix} \frac{f_x}{z^{l^n}_{Fj}} & 0 & \frac{-x^{l^n}_{Fj} \cdot f_x}{(z^{l^n}_{Fj})^2} \\ 0 & \frac{f_y}{z^{l^n}_{Fj}} & \frac{-y^{l^n}_{Fj} \cdot f_y}{(z^{l^n}_{Fj})^2} \end{pmatrix} \cdot \mathbf{R}^{Fj}_{m^l} \cdot \begin{bmatrix} [P^{l^n}_{m^l}]_\times & -\mathbf{I}_3 \end{bmatrix}_{3\times6} \end{aligned} \tag{3}$$

D. Camera Groups Bundle Adjustment

In this section, the extrinsics among cameras are used as prior information, and we introduce the camera groups coordinate system denoted by $(\cdot)_{gj}$. Frames here are defined as $F = \{F^{1k}, F^{2k}, \cdots, F^{jk}, \cdots F^{Jk}\}$, F^{jk} denotes an image captured by k-th camera at j-th location, where k is the index of camera in the camera groups. For ease of presentation, this section uses object points to derive the camera groups bundle adjustment. $X^i_{Fjk} = [x^i_{Fjk}, y^i_{Fjk}, z^i_{Fjk}]^\top$ denotes the corresponding points of X^i in $(\cdot)_{Fjk}$. X^i_{gj} denotes the corresponding points of X^i in $(\cdot)_{gj}$. We also define the set of indices for object points captured by frame F^{jk} as $\mathbb{C}^{jk}$, and the measurement of image coordinate for the i-th object point in the jk-th frame is defined as x^{ijk} for $i \in \mathbb{C}^{jk}$ and

$j \in \{0,1,...,J\}, k \in \{0,1,...,N_{cam}\}$, where N_{cam} is the number of cameras in the camera groups. The transformation matrix from jk-th frame to its camera groups is $\mathbf{T}_{F_{jk}}^{g_j}$.

In camera groups bundle adjustment, the projection procedure from X^i to x^{ijk} is $x^{ijk} = \pi_{\mathbf{K}}(\mathbf{T}_{g_j}^{F_{jk}} \mathbf{T}_w^{g_j} X^i)$. By adding small perturbation variables $\delta\xi^g$ to the camera groups pose $\mathbf{T}_w^{g_j}$, projection error χ^{ig} can be defined as

$$\begin{aligned} \chi^{ijk} &= x^{ijk} - \pi_{\mathbf{K}}(X_{F_{jk}}^k) \\ X_{F_{jk}}^k &= \mathbf{T}_{g_j}^{F_{jk}} \exp(\delta\hat{\xi}^g) \mathbf{T}_w^{g_j} X^i \end{aligned} \tag{4}$$

We can derive the partial derivatives of $P_{F_{jk}}^i$ with respect to the camera groups pose $\delta\xi^g$, which is given as

$$\begin{aligned} \frac{\partial X_{F_{jk}}^k}{\partial \delta\xi^g} &= \lim_{\delta\xi^g \to 0} \frac{\mathbf{T}_{g_j}^{F_{jk}} \exp(\delta\hat{\xi}^g) \mathbf{T}_w^{g_j} X^i - \mathbf{T}_{g_j}^{F_{jk}} \mathbf{T}_w^{g_j} X^i}{\delta\xi^g} \\ &= \lim_{\delta\xi^g \to 0} \frac{\mathbf{T}_{g_j}^{F_{jk}} \exp(\delta\hat{\xi}^g) \mathbf{T}_w^{g_j} X^i}{\delta\xi^g} = \lim_{\delta\xi^g \to 0} \frac{\mathbf{T}_{g_j}^{F_{jk}} (\mathbf{I} + \delta\hat{\xi}^g) X_{g_j}^i}{\delta\xi} \\ &= \mathbf{T}_{g_j}^{F_{jk}} \times \begin{bmatrix} [-X_{g_j}^i]_\times & \mathbf{I}_3 \\ 0 & 0 \end{bmatrix} = \mathbf{R}_{g_j}^{F_{jk}} \cdot \begin{bmatrix} [-X_{g_j}^i]_\times & \mathbf{I}_3 \end{bmatrix}_{3\times 6} \end{aligned} \tag{5}$$

Finally, the partial derivatives of χ^{ijk} with respect to marker pose increment $\delta\xi^g$ is

$$\begin{aligned} \frac{\partial \chi^{ijk}}{\partial \delta\xi^g} &= \frac{\partial \chi^{ijk}}{\partial X_{F_{jk}}^k} \cdot \frac{\partial X_{F_{jk}}^k}{\partial \delta\xi^g} \\ &= \begin{pmatrix} \frac{f_x}{z_{F_{jk}}^i} & 0 & \frac{-x_{F_{jk}}^i \cdot f_x}{(z_{F_{jk}}^i)^2} \\ 0 & \frac{f_y}{z_{F_{jk}}^i} & \frac{-y_{F_{jk}} \cdot f_y}{(z_{F_{jk}}^i)^2} \end{pmatrix} \cdot \mathbf{R}_{g^j}^{F_{jk}} \cdot \begin{bmatrix} [-X_{g_j}^i]_\times & \mathbf{I}_3 \end{bmatrix}_{3\times 6} \end{aligned} \tag{6}$$

E. Measuring Observation Weight for Information Matrix

Insufficient observation for the planner marker is a significant challenge in pose estimation due to the ambiguity in the marker pose [7] [11]. In this study, we propose an effective method for computing the information matrix for different marker observations. We use IPPE [23] for initial marker pose estimation, which obtains two pose estimations with corresponding reprojection errors. In some atypical scenarios, such as an oblique observation angle to marker and long observation distance, the reprojection error of solutions may be very similar, making it difficult to determine the correct one. In this section, we consider the estimated marker distance d, the angle of the light axis θ, and the distribution of the two rotation solutions (r_1, r_2) from [23] as

$$\begin{aligned} w(d,\theta,r_1,r_2) = \exp\Bigg(& -\lambda_1 \left(\frac{d}{d_{\max}}\right)^2 - \lambda_2 \left(\frac{\theta}{\theta_{\max}}\right)^2 \\ & + \lambda_3 \cdot \exp\left(-\frac{|r_1 - r_2|^2}{2\varepsilon^2}\right)\Bigg) \end{aligned} \tag{7}$$

where θ_{max} represents the maximum angle of the light axis, and d_{max} denotes the maximum distance. From this equation, smaller d and θ, as well as higher similarity of two rotations from IPPE [23] will result in a lower weight. In all experiments, we set hyper parameters $\varepsilon = 0.1$, $d_{max} = 15\,\text{m}$, and $\theta_{max} = 90^\circ$. The factors λ_1, λ_2, and λ_3 will be manually set in the experiments. After this, the information matrix of observation is defined as

$$\mathbf{I_{obs}} = w(d,\theta,r_1,r_2) \cdot \mathbf{E_{2\times 2}} \tag{8}$$

where $\mathbf{E}_{2\times 2}$ is an identity matrix with a dimension of 2×2. This weighted information matrix will be used in the factor graph optimization later.

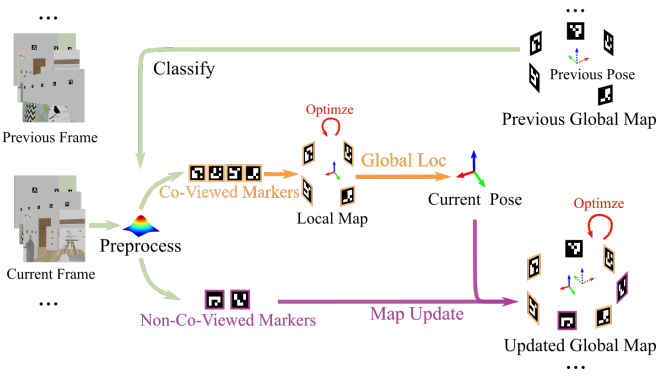


Fig. 3. The overview of our incremental marker-CG SfM framework is illustrated. The framework is divided into three parts: Preprocessing, Global Localization, and Incremental Map Update. Marker recognition and information confidence calculation are integrated in the preprocessing part. For clear statement, map initialization is not included in this overview.

F. Incremental Marker-CG SfM Framework

In our overall mapping framework, we utilize PnP [23] in the front-end to perform initial marker pose estimation and initial camera groups localization. To address vast scenarios effectively, we employ factor graph optimization, which exhibits good convergence and robustness, making it suitable for massive SfM optimization in our paper. Therefore, based on Section III-C and Section III-D, the back-end based on factor graph optimization is designed. The overall framework is divided into three stages: map initialization (Section III-F.1), global visual localization (Section III-F.2) and incremental map update (Section III-F.3). The input is a collection of images, and the output are the reconstructed marker map and recovered camera poses.

1) Map Initialization: Our algorithm randomly selects one frame F_j (F_{jk} if the proposed method takes input from camera groups, for simplicity, 'if *camera groups*' will be used subsequently) from the input image collections and sets its pose to a identity matrix. This frame remains fixed during optimization and is treated as the origin. Then, we recognize all markers in F_j or F_{jk} and use the predefined $P_{m_l}^{l_n}$ and detected marker pixel to calculate the initial pose of each marker in the world coordinate using PnP [23].

After recovering all markers in F_j or F_{jk} to the world coordinate, an initial global factor map G_M consisting of a frame with a unit matrix pose and multiple markers can be established. The subsequent incremental marker SfM are performed on G_M. The initial frame can be randomly selected, and in our experiment, the frame with the highest number of recognized markers is chosen.

2) Optimization-based Global Camera Localization: We optimize the 6-DOF pose of camera in the world coordinate by constructing a local factor graph G_L, and bundle

adjustment optimization in Section III-C and Section III-D are used to average errors. Specifically, as is shown in Fig. 3, we classify recognized markers M_{det} in F_j or F_{jk} as *co-viewed markers* M_c and *non-co-viewed markers* M_{nc} based on the rule whether it has been added to the map G_M or not. During the creation of the marker map G_L, we copy the accurate pose of M_c from G_M to construct a factor graph G_L. After the creation of the G_L, all image observations $p_{m^l}^{l^n}$ of markers M_{det} will be traversed and constraints between marker pose $\mathbf{T}_w^{m^l}$ and camera pose $\mathbf{T}_w^{F^j}$ or $\mathbf{T}_w^{F^{jk}}$ will be added to G_L. Notice that, the marker poses $\mathbf{T}_w^{m^l}$ in the factor graph G_L are fixed, and only the camera pose $\mathbf{T}_w^{F^j}$ or $\mathbf{T}_w^{F^{jk}}$ can be optimized. To reduce noises in the optimization, we introduce Huber Loss [24], [25] σ to eliminate observations with errors greater than a certain threshold. The objective function $\mathbf{L}_{loc}$ for global localization is defined as

$$\mathbf{L}_{loc} = \min_{\delta\xi^g} \sum_{j=0}^{J} \sum_{k=0}^{N_{cam}} \sum_{i \in \mathbb{C}^{jk}} \sigma ||(\chi^{ijk})^{\mathbf{T}} \cdot \mathbf{I_{obs}} \cdot \chi^{ijk}||^2 \tag{9}$$

where $\mathbf{I_{obs}}$ is the corresponding information matrix of observation error χ^{ijk} calculated in Section III-E.

In the experiment, to ensure the convergence of the optimization process, we select the marker index M_{min} based on the rule that the reprojection error is minimized when solving for $\mathbf{T}_w^{m^l}$ using its observations, which is defined as

$$\mathbf{T}_w^{m^l}(\textbf{initial guess}) = \underset{\mathbf{T}_w^{m^l} \in \mathbf{T}_w^{m^L}}{\arg\min} \sum_{n=0}^{4} \left\| x_{m^l}^{l^n} - \pi_{\mathbf{K}}(\mathbf{T}_w^{m^l} X_{m^l}^{l^n}) \right\|_2^2 \tag{10}$$

We use $\mathbf{T}_w^{F^j}$ or or $\mathbf{T}_w^{F^{jk}}$ calculated from $\mathbf{T}_w^{m^l}(\textbf{initial guess})$ as the initial optimization guess for global localization. In PytagMapper [8], the initial optimization guess is calculated by all markers observation. Experimental results demonstrate that initial guess is the underlying factor of localization failure. The detailed procedures are shown in Algorithm 1.

3) Map Update and Global Factor Graph Optimization: After completing the global localization of a newly added frame F_j or F_{jk}, we need to traverse all detected markers M_{det} in F_j or F_{jk}.

For *non-co-viewed markers* M_{nc}, we update them to the map G_M using PnP [23]. Noted that, in the case of camera groups, after solving the relative pose between the marker and the k-th observed camera, it is necessary to multiply it by the extrinsics of the camera groups to this k-th observed camera. For all observed markers M_{det}, we add observation constraints to the factor graph G_M. Finally, an optimization is performed. We define the set of markers indices captured by frame F^{jk} as $\mathbb{C}^{jkm}$, and the objective function $\mathbf{L}_{SfM}$ for Marker-CG SfM is defined as

$$\begin{aligned}\mathbf{L}_{SfM} = \min_{\{\delta\xi^m, \delta\xi^g\}} \sum_{j=0}^{J} \sum_{k=0}^{N_{cam}} \sum_{n=0}^{4} \sum_{i \in \mathbb{C}^{jkm}} \sigma ||(\chi^{ijk})^{\mathbf{T}} \cdot \mathbf{I_{obs}} \cdot \chi^{ijk} \\ + (\chi^{jmn})^{\mathbf{T}} \cdot \mathbf{I_{obs}} \cdot \chi^{jmn}||^2\end{aligned} \tag{11}$$

where $\mathbf{I_{obs}}$ is the corresponding information matrix of observation error χ^{ijk} and χ^{jmn}. The incremental map update and optimization are shown in Algorithm 2 and Fig. 3.

Algorithm 1: Optimization-based Global Visual Localization with Reconstructed Marker Map

Input: G_M, F_j or F_{jk}, M_{det}, M_{co}, M_{nc}, M_{min}
Output: 6-DOF Pose $\mathbf{T}_w^{F^j}$ or $\mathbf{T}_w^{F^{jk}}$) of F_j or F_{jk}

NEW factor graph G_L
if *monocular* **then**
  $\mathbf{T}_w^{F^j}$(**initial guess**) =**PnP** ($p_{m^l}^{M^n_{min}}$, $P_{m^l}^{M^n_{min}}$)
  $\mathbf{T}_w^{F^j}$(**initial guess**) $\rightarrow F_j \rightarrow G_L$
else if *camera groups* **then**
  $\mathbf{T}_w^{F^{jk}}$(**initial guess**) $= \mathbf{T}_{F^{jk}}^{g^j} \cdot$ **PnP** ($p_{m^l}^{M^n_{min}}$, $P_{m^l}^{M^n_{min}}$)
  $\mathbf{T}_w^{F^{jk}}$(**initial guess**) $\rightarrow F_{jk} \rightarrow G_L$
end
for M **in** M_c **do**
  $\mathbf{T}_w^{m^l}$ **from** $G_M \rightarrow M \rightarrow G_L$
  ADDCONSTRAINTS($p_{m^l}^{M^n}$, $P_{m^l}^{M^n}$)
end
OPTIMIZE G_L
RETURN $\mathbf{T}_w^{F^j}$ or $\mathbf{T}_w^{F^{jk}}$ from G_L

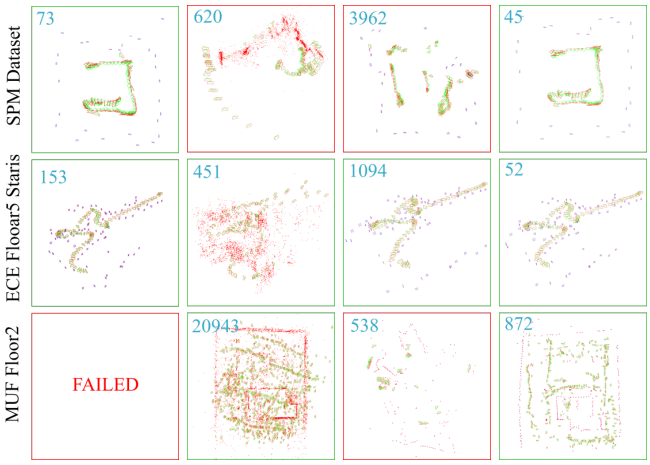
Algorithm 2: Incremental Map Update and Factor Graph Optimization

Input: G_M, M_{det}, F_j, Set($\mathbf{T}_{F^{jk}}^{g^j}$), N_{cam}, *isinited*=false
if *monocular* **then**
  N_{cam} = 1, Set($\mathbf{T}_{F^{jk}}^{g^j}$) = [$\mathbf{E}$]
end
for k **in** N_{cam} **do**
  if !*isinited* **then**
    $\mathbf{T}_w^{F^{jk}} = \mathbf{E}_{4\times4}$, M_{nc} = M_{det}, *isinited*=true;
  else
    CLASSIFY M_c and M_{nc}
    $\mathbf{T}_w^{F^{jk}}$ = ALG.GLOBAL LOCALIZATION
  end
  for M **in** M_{nc} **do**
    $\mathbf{T} = \mathbf{T}_{F^{jk}}^{g^j} \cdot$ PNP($p_{m^l}^{M^n}$, $P_{m^l}^{M^n}$)
    $\mathbf{T}_w^{F^{jk}} \cdot \mathbf{T} \rightarrow M \rightarrow G_M$
    ADDCONSTRAINTS($p_{m^l}^{M^n}$, $P_{m^l}^{M^n}$, $\mathbf{T}_{F^{jk}}^{g^j}$)
  end
end
OPTIMIZE G_M

IV. EXPERIMENT

Our algorithm is implemented on the ROS platform [26], and the back-end relies on the optimization library Ceres-Solver [27]. All baselines are implemented using the same platform, specifically the Intel i9-13900HX Processor (5.4GHz). In our experiments, we assume that the camera intrinsics and camera groups extrinsics are calibrated via Kalibr [28].

Firstly, we evaluate the performance of our algorithm on public datasets, including SPM-SLAM [6] and Degol [7] (in



(a) MarkerMapper (b) Degol (c) PytagMapper (d) Ours

Fig. 4. 3D reconstruction results in public dataset [6] [7]. We select three challenging sequences in this comparison experiment. Reconstruction results show that our algorithm reconstructs all sequences successfully. The number in the upper left corner of the image is the time consumed (in seconds) with the algorithm. The green boxes indicate the complete results and the red boxes indicate failure cases.

Section IV-A). The input data for Degol [7] are unordered sets of images. For SPM-SLAM [6], the datasets are videos, and we sample one frame every ten frames from the video.

Furthermore, we have constructed a customed calibration room outfitted with sevaral markers, and multiple camera systems are employed to capture this environment. The reconstruction result will be presented in Section IV-B.

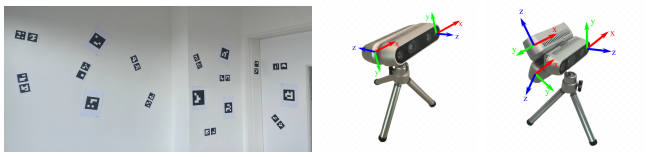
Due to the unreliability of ground truth in real-world scenarios, the results of the above experiments are qualitative. In this paper, we quantitatively evaluate the various algorithm on our simulated datasets with different marker sizes and camera groups in Section IV-C and Section IV-D. Finally, an ablation study regarding the computation of the information matrix is discussed in Section IV-E.

A. Reconstuction Results on Public Monocular Datasets

To begin with, we evaluate the performance of our framework on publicly available datasets [6], [7], which are based on ArucoTag [21] and AprilTag [29], [30]. As shown in Fig. 4, our method achieves excellent results in terms of reconstruction accuracy and robustness. Moreover, our algorithm is highly efficient, as it only utilizes marker information and does not require time-consuming feature point detection and registration between image pairs. Consequently, our approach can significantly reduce computational time, even on large datasets containing up to 896 images, which can be processed within 15 minutes.

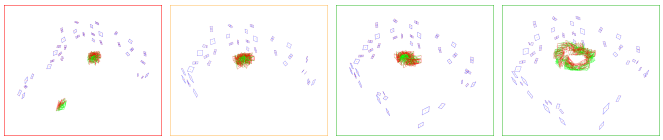
B. Reconstruction Results on Self Captured Real Scenario

A physical calibration room is designed with multiple size markers and two to three Intel RealSense d435i cameras are utilized to capture the environment. The layout of the camera groups and the calibration room are shown in Fig. 5. The reconstruction results of the calibration room, including the camera groups and multiple size markers, are presented in Fig. 6. The results show that in this real-world scenario,



(a) Calibration Room (b) CG-180 (c) CG-120

Fig. 5. (a) A real scene equipped with multiple size markers. (b) Two cameras are coupled back to back at a 180° angle. (c) Three cameras are coupled at a 120° angle.



(a) PytagMapper (b) Mono (c) CG-120 (d) CG-180

Fig. 6. 3D reconstruction results of the monocular camera and the camera groups in real calibration room. The green boxes and red boxes serve the same meanings as described in the last section. The yellow boxes indicate incomplete but correct results. Baselines [5] [7] cannot address the situation with multiple marker sizes. Both the baseline [8] (a) and our monocular algorithm (b) have encountered limitations or incompleteness because of the narrow visual FoV, which represents a key bottleneck for monocular algorithms. (c) and (d) demonstrate the successful reconstructions via out camera groups SfM algorithm.

the real scene can be efficiently reconstructed using different camera groups configurations.

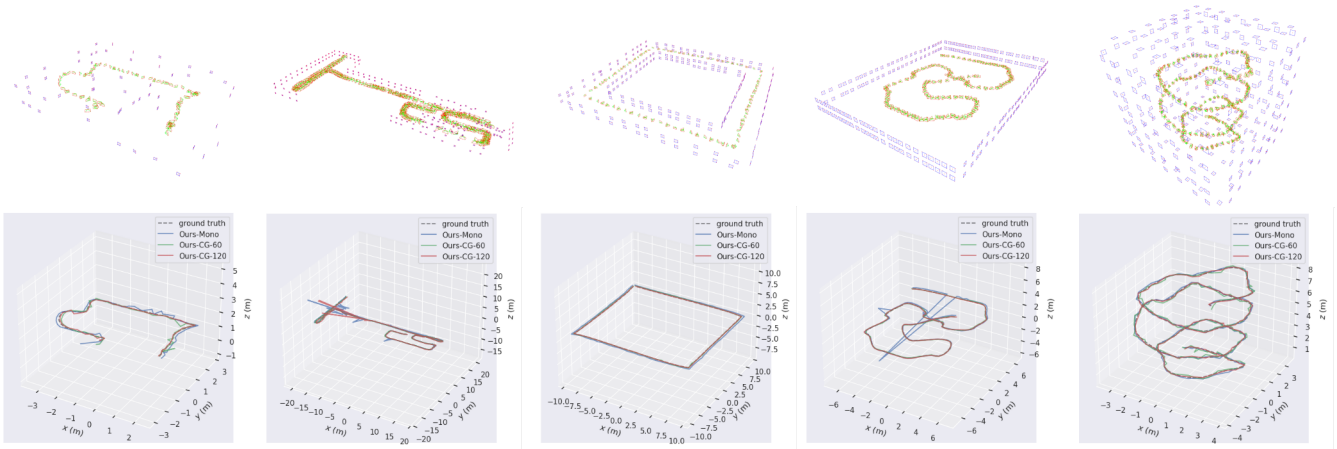
C. Reconstruction Results on Proposed Datasets of Same Marker Size

Fig. 7(a) and Fig. 7(b) show the reconstruction comparisons of existing algorithms and our algorithm in the proposed datasets. Table II shows the quantitative metrics of each algorithm in different sequences. Absolute Trajectory Error (ATE) of pose rotation (in degrees) and translation (in meters) demonstrate that our method achieves the minimum RMSE of ATE error not only with input in the form of camera groups but also with input in the form of monocular. Because our simulation scenes are relatively vast, other algorithms are unable to reconstruct successfully or attain low accuracy. Degol's algorithm [7] cannot complete the reconstruction in the vast and texture-less scenarios, so it is not listed. To ensure fairness in monocular camera reconstruction, we input all images from camera groups for validation.

TABLE II

PERFORMANCE COMPARISON ON PROPOSED DATASETS OF UNIQUE MARKER SIZE AMONG DIFFERENT METHODS

Sequence	Framework	Marker		Camera		Time (s)
		Rotation	Translation	Rotation	Translation	
Indoor Room 1	PytagMapper	3.231	0.646	8.101	1.431	1230
	Ours-Mono	0.912	0.096	4.362	0.364	154
	Ours-CG-60	**0.736**	**0.076**	2.009	0.150	205
	Ours-CG-120	0.753	0.085	**0.692**	**0.069**	201
Indoor Room 2	PytagMapper	3.560	0.851	13.843	2.428	1620
	Ours-Mono	**0.799**	**0.085**	1.834	0.224	185
	Ours-CG-60	0.821	0.088	1.691	0.253	286
	Ours-CG-120	0.998	0.102	**0.986**	**0.109**	281
Swimming Pool	PytagMapper	×	×	×	×	×
	Ours-Mono	6.930	2.211	5.313	2.288	2778
	Ours-CG-60	1.221	0.113	2.517	1.952	1730
	Ours-CG-120	**0.921**	**0.109**	**0.912**	**0.105**	1555



(a) Indoor Room 1 (b) Swimming Pool (c) Corridor (d) Warehouse (e) Calibration Room

Fig. 7. The 3D reconstruction results and ATE comparisons on proposed synthetic marker datasets. The first row of images is a 3D reconstruction of CG-120, and the second row is a comparison of different methods with the ground truth trajectory.

D. Reconstruction Results on Proposed Datasets of Different Marker Size

Our method is capable of handling multi-size marker and camera group inputs, as demonstrated in Fig. 7(c), Fig. 7(d), Fig. 7(e), and Table III, where we showcase the reconstruction results and performance on various scenarios. Notably, improved PytagMapper [8] fails to complete the reconstruction in vast scenarios, and thus it is not included in the table and figures.

Table II and Table III demonstrate that the implementation of camera groups can significantly improve reconstruction accuracy compared to monocular camera in vast scenes, such as warehouse and swimming pool. Furthermore, if the arrangement of camera groups is deployed to increase the FoV and observe more information (specifically, changing CG-60 to CG-120), the improvement in reconstruction accuracy will be even greater.

TABLE III

PERFORMANCE COMPARISON ON PROPOSED DATASETS OF DIFFERENT MARKER SIZES AMONG DIFFERENT METHODS

Sequence	Framework	Marker		Camera		Time (s)
		Rotation	Translation	Rotation	Translation	
Corridor	Ours-Mono	0.923	0.096	0.970	0.097	1221
	Ours-CG-60	1.112	0.113	1.308	0.146	773
	Ours-CG-120	**0.692**	**0.041**	**0.659**	**0.045**	907
Warehouse	Ours-Mono	1.269	0.127	8.765	1.335	2521
	Ours-CG-60	0.771	0.071	0.927	0.091	2134
	Ours-CG-120	**0.604**	**0.062**	**0.685**	**0.071**	3129
Calibration Room	Ours-Mono	1.021	0.092	1.076	0.104	2293
	Ours-CG-60	0.544	0.052	1.321	0.168	521
	Ours-CG-120	**0.405**	**0.039**	**0.812**	**0.083**	651

To summarize, our proposed approach outperforms algorithms [5], [8] that solely rely on markers for map reconstruction in terms of accuracy. Additionally, compared to algorithms [7] that use both markers and natural feature points, our method is more time-efficient. The use of camera groups, as opposed to a monocular camera, increases the observable environment information, which can overcome the limitations of insufficient reconstruction accuracy and localization failure in some cases (as illustrated in the failure case of monocular camera in Fig. 7(d)). For more experimental results, please refer to the video on Github.

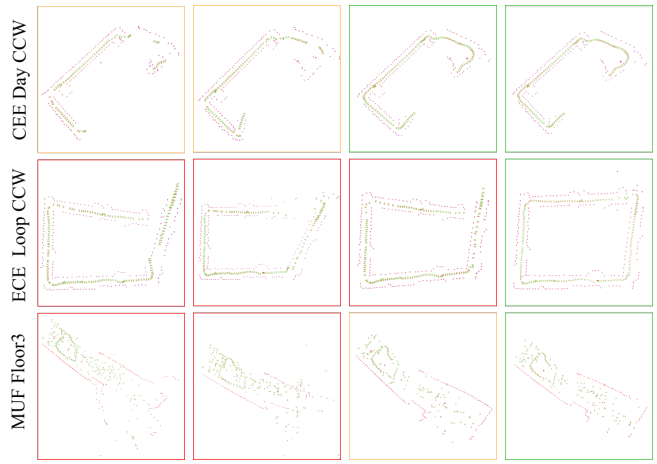
E. Ablation Comparison with Different Information Matrices

The qualitative comparison in public monocular datasets is shown in Fig. 8, while the quantitative comparison on our proposed synthetic datasets is demonstrated in Table IV. Here, L ($\lambda_1 = \lambda_2 = \lambda_3 = 0$) represents the parameter setting when all information matrices are identical for all observations. The hyper parameter setting denoted as L_d ($\lambda_1 = 1,\ \lambda_2 = \lambda_3 = 0$) is obtained when only the distance factor is considered in the information matrix. Similarly, $L_{d/\theta}$ ($\lambda_1 = 1,\ \lambda_2 = 5,\ \lambda_3 = 0$) represents the hyper parameter setting when both the distance and observation angle are taken into account in the information matrix. Finally, L_{all} ($\lambda_1 = 1,\ \lambda_2 = 5,\ \lambda_3 = 0.1$) is the setting when all factors are considered. It is evident that in certain sequences, disabling information estimation can lead to reconstruction failures. This emphasizes the importance of the calculation algorithm of marker observation information matrix, which can potentially serve as a supportive approach to address the ambiguity in the marker pose.

TABLE IV

ABLATION COMPARISON ON CALCULATION OF INFORMATION MATRIX INTRODUCED IN SECTION III-E.

Sequence	Parameter	Marker		Camera	
		Rotation	Translation	Rotation	Translation
Indoor Room 1	L	1.209	0.098	1.158	0.094
	L_d	1.239	0.093	1.009	0.084
	$L_{d/\theta}$	1.253	0.108	1.092	0.089
	L_{all}	**0.753**	**0.085**	**0.692**	**0.069**
Warehouse	L	1.250	0.101	1.665	0.143
	L_d	1.271	0.112	1.302	0.119
	$L_{d/\theta}$	0.994	0.089	1.085	0.098
	L_{all}	**0.604**	**0.062**	**0.685**	**0.071**
Calibration Room	L	1.088	0.092	1.201	0.149
	L_d	1.283	0.129	1.420	0.235
	$L_{d/\theta}$	0.545	0.048	1.193	0.092
	L_{all}	**0.405**	**0.039**	**0.812**	**0.083**



(a) L (b) L_d (c) $L_{d/\theta}$ (d) L_{all}

Fig. 8. The impact of different information matrices on public monocular datasets is illustrated. The boxes serve the same meanings as described in the last section. In the sequence shown above, as the weight calculation factors are gradually considered in the calculation, the reconstructed result is gradually completed and proved to be a success.

V. Conclusion

In this paper, we propose a novel incremental SfM framework that leverages PnP in the front-end and customized bundle adjustment in the back-end, which achieves accurate and robust performance in challenging environments with varying marker sizes and multiple cameras. Additionally, we propose a dataset with ground truth pose labels for marker-based SfM. To evaluate the reconstruction accuracy of our approach, we conduct experiments in public datasets and proposed datasets. Experiments demonstrate that our method achieves high accuracy and fast speed. Overall, our approach provides a promising solution to the challenges of marker-based SfM, and our dataset with ground truth pose labels can serve as a valuable resource for future research in this field.

References

[1] J. Yuan, S. Zhu, K. Tang, and Q. Sun, "ORB-TEDM: An RGB-D slam approach fusing ORB triangulation estimates and depth measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–15, 2022.

[2] C. Sweeney, T. Hollerer, and M. Turk, "Theia: A fast and scalable structure-from-motion library," in *Proceedings of the 23rd ACM International Conference on Multimedia*, 2015, pp. 693–696.

[3] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski, "Building Rome in a day," *Communications of the ACM*, vol. 54, no. 10, pp. 105–112, 2011.

[4] S. Agarwal, Y. Furukawa, N. Snavely, B. Curless, S. M. Seitz, and R. Szeliski, "Reconstructing Rome," *Computer*, vol. 43, no. 6, pp. 40–47, 2010.

[5] R. Muñoz-Salinas, M. J. Marín-Jimenez, E. Yeguas-Bolivar, and R. Medina-Carnicer, "Mapping and localization from planar markers," *Pattern Recognition*, vol. 73, pp. 158–171, 2018.

[6] R. Munoz-Salinas, M. J. Marin-Jimenez, and R. Medina-Carnicer, "SPM-SLAM: Simultaneous localization and mapping with squared planar markers," *Pattern Recognition*, vol. 86, pp. 156–171, 2019.

[7] J. DeGol, T. Bretl, and D. Hoiem, "Improved structure from motion using fiducial marker matching," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 273–288.

[8] M. Markisus, "Pytagmapper," https://github.com/markisus/pytagmapper, 2022.

[9] S.-F. Ch'ng, N. Sogi, P. Purkait, T.-J. Chin, and K. Fukui, "Resolving marker pose ambiguity by robust rotation averaging with clique constraints," pp. 9680–9686, 2020.

[10] R. Munoz-Salinas and R. Medina-Carnicer, "UcoSLAM: Simultaneous localization and mapping by fusion of keypoints and squared planar markers," *Pattern Recognition*, vol. 101, p. 107193, 2020.

[11] Z. Jia, Y. Rao, H. Fan, and J. Dong, "An efficient visual SfM framework using planar markers," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–12, 2023.

[12] S. Zhu, R. Zhang, L. Zhou, T. Shen, T. Fang, P. Tan, and L. Quan, "Very large-scale global SfM by distributed motion averaging," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4568–4577, 2018.

[13] W. Burger and M. J. Burge, "Scale-invariant feature transform (SIFT)," *Digital Image Processing: An Algorithmic Introduction*, pp. 709–763, 2022.

[14] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 756–770, 2004.

[15] R. I. Hartley, "In defense of the eight-point algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 580–593, 1997.

[16] J. Ortiz, T. Evans, and A. J. Davison, "A visual introduction to Gaussian Belief Propagation," *arXiv preprint arXiv:2107.02308*, 2021.

[17] S. Urban and S. Hinz, "Multicol-SLAM - a modular real-time multi-camera slam system," *arXiv preprint arXiv:1610.07336*, 2016.

[18] K. Eckenhoff, P. Geneva, and G. Huang, "MIMC-VINS: A versatile and resilient multi-IMU multi-camera visual-inertial navigation system," *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1360–1380, 2021.

[19] M. Abate, A. Schwartz, X. I. Wong, W. Luo, R. Littman, M. Klinger, L. Kuhnert, D. Blue, and L. Carlone, "Multi-camera visual-inertial simultaneous localization and mapping for autonomous valet parking," *arXiv preprint arXiv:2304.13182*, 2023.

[20] B. O. Community, *Blender - a 3D Modelling and Rendering Package*, http://www.blender.org, Blender Foundation, Blender Institute, Amsterdam, 2013.

[21] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014.

[22] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 573–580.

[23] T. Collins and A. Bartoli, "Infinitesimal plane-based pose estimation," *International Journal of Computer Vision*, vol. 109, no. 3, pp. 252–286, 2014.

[24] 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.

[25] M. I. Lourakis and A. A. Argyros, "SBA: A software package for generic sparse bundle adjustment," *ACM Transactions on Mathematical Software*, vol. 36, no. 1, pp. 1–30, 2009.

[26] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "ROS: an open-source robot operating system," in *Proceedings of the IEEE International Conference on Robotics and Automation Workshop on Open Source Robotics*, Kobe, Japan, 2009.

[27] S. Agarwal, K. Mierle, and T. C. S. Team, "Ceres Solver," 10 2023. [Online]. Available: https://github.com/ceres-solver/ceres-solver

[28] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 1280–1286.

[29] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2011, pp. 3400–3407.

[30] J. Wang and E. Olson, "AprilTag 2: Efficient and robust fiducial detection," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2016, pp. 4193–4198.