Hierarchical Visual SLAM based on Fiducial Markers

Ali Tourani[1], Hriday Bavle[1], Jose Luis Sanchez-Lopez[1], Rafael Muñoz Salinas[2], and Holger Voos[1]

***Abstract*— Fiducial markers can encode rich information about the environment and aid Visual SLAM (VSLAM) approaches in reconstructing maps with practical semantic information. Current marker-based VSLAM approaches mainly utilize markers for improving feature detection in low-feature environments and/or incorporating loop closure constraints, generating only low-level geometric maps of the environment prone to inaccuracies in complex environments. To bridge this gap, this paper presents a VSLAM approach utilizing a monocular camera and fiducial markers to generate hierarchical representations of the environment while improving the camera pose estimate. The proposed approach detects semantic entities from the surroundings, including walls, corridors, and rooms encoded within markers, and appropriately adds topological constraints among them. Experimental results on a real-world dataset demonstrate that the proposed approach outperforms a traditional marker-based VSLAM baseline in terms of accuracy, despite adding new constraints while creating enhanced map representations. Furthermore, it shows satisfactory results when comparing the reconstructed map quality to the one reconstructed using a LiDAR SLAM approach.**

I. Introduction

The primary advantage of vision sensors in Visual SLAM (VSLAM) systems is that they need low-cost hardware to supply rich visual and semantic information from surroundings for various tasks [1]. Semantic data, which refers to high-level information acquired from the environment, make VSLAM tasks more robust and expand the range of applications that can employ the reconstructed maps [2], [3]. In this regard, utilizing fiducial markers is one of the possible approaches to encoding semantic information into the environment [4]. They can assist VSLAM frameworks by providing accurate pose estimation, supplying reliable features in low-texture environments, and enabling loop closure detection. Recent works such as [5] and [6] propose VSLAM approaches using fiducial markers but do not encode them with meaningful semantic information, creating purely geometric map representations leading to inaccuracies in camera pose estimates and the generated environmental map in the presence of complex environments.

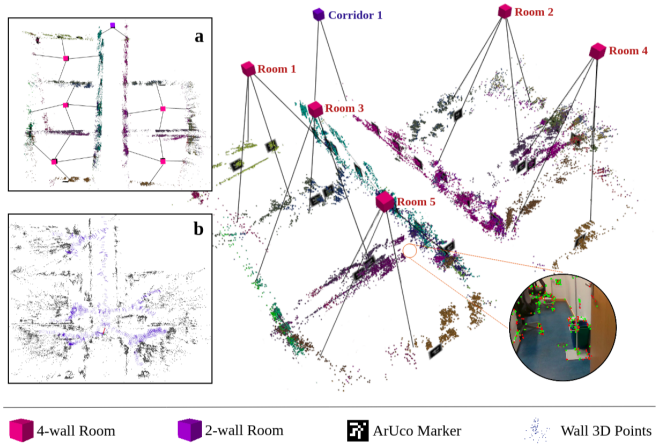

Fig. 1: The final reconstructed map of the environment using the proposed method in a hierarchical representation: a) the top view of the reconstructed map represented in 2D, b) keypoints and robot trajectory records.

To fully leverage the potential of fiducial markers in accurately identifying both the semantic elements and their topological relationships while generating maps, this paper proposes a VSLAM framework for monocular cameras that utilizes the data encoded in fiducial markers for enhanced map reconstruction, adding abstract semantic elements to the final map along with their topological relationships. The system is built upon UcoSLAM [5], which employs ArUco [7] markers and visual keypoints extracted from natural landmarks reconstructing the geometric map of the environment. Additionally, inspired by *S-Graphs* [8], [9], the proposed approach adds extraordinary information in the form of walls, corridors, and rooms considering the data encoded in the ArUco markers. A sample map reconstructed by the proposed approach and its hierarchical representation is demonstrated in Fig. 1.

Herewith the main contributions of the paper are summarized below:

- An extension of marker-based VSLAM to reconstruct environmental maps with high-level semantic features,
- The design of novel geometric constraints, namely *marker-to-wall* and *wall-to-room*, to improve the map quality and reduce camera localization errors,
- Validation of the proposed method using a real-world indoor dataset showing improved performance over marker-based VSLAM baseline.

[1]Authors are with the Automation and Robotics Research Group, Interdisciplinary Centre for Security, Reliability, and Trust (SnT), University of Luxembourg, Luxembourg. Holger Voos is also associated with the Faculty of Science, Technology, and Medicine, University of Luxembourg, Luxembourg. {ali.tourani, hriday.bavle, joseluis.sanchezlopez, holger.voos}@uni.lu

[2]Author is with the Department of Computer Science and Numerical Analysis, Rabanales Campus, University of Córdoba, Spain. rmsalinas@uco.es

This work was funded by the Institute of Advanced Studies (IAS) of the University of Luxembourg (project TRANSCEND), the European Commission Horizon2020 research and innovation program under the grant agreement No 101017258 (SESAME), and the Luxembourg National Research Fund (FNR) 5G-SKY project (ref. C19/IS/13713801).

II. Related Works

The authors of this paper presented a comprehensive survey on diverse VSLAM state-of-the-art works and studied their trade-offs in [10]. Accordingly, approaches like UcoSLAM [5] and TagSLAM [6] use the capability of fiducial markers for SLAM tasks. However, they cannot obtain semantic information from the environment for an improved map representation. Works like PL-SLAM [11] and LIFT-SLAM [12] show accurate feature detection and tracking, but they are computationally intensive. S^{3}LAM [13] and YOLO-SLAM [14] use Convolutional Neural Network (CNN)-based semantic segmentation of generic objects and structures but suffer from performance degradation in recognizing small or non-regular objects. In contrast with the mentioned works, the proposed approach creates a map of the environment by utilizing semantic data encoded within makers instead of employing any object detector. Accordingly, new constraints are estimated based on mathematical modeling obtained from keypoints, which do not significantly affect the performance of the system.

III. Proposed Method

In order to develop a VSLAM framework with richer reconstructed maps, the proposed approach utilizes UcoSLAM as the baseline and modifies its components to be empowered with a semantic data analysis procedure. The mentioned modifications enable the detection of *walls* and different types of *rooms* as two semantic concepts in the environment using ArUco markers. This method also aspires to employ visual data to represent the environment and the robot's pose in a single optimizable graph with an approach comparable to *S-Graphs* [15].

A. Overview

The pipeline of the proposed method at time t can be referred to four main coordinate systems: the odometry frame of reference O, the camera coordinate system C_t, the marker coordinate system M_t, and the global coordinate system G_t. As the primary sensor of the system, a monocular camera acquires a set $\mathsf{F} = \{\mathsf{f}\}$ of frames $\mathsf{f} = \{t, \mathsf{I}, \}$, where $\mathsf{T} \in SE(3)$ is the camera's pose obtained from the transformation of C_t to G_t, and is the set of camera intrinsic parameters. By processing each camera frame using Oriented FAST and Rotated BRIEF (ORB) keypoint detector and feature extractor, a set of keypoints $\mathsf{g} = \{l, u, \mathsf{d}\}$ are extracted in which l is the subsampling level of the image, u is the pixel coordinates for upsampling w.r.t. the first level, and $\mathsf{d} = (d_1 \dots d_n) | d_i \in [0, 1]$ is the descriptor vector with length n. Accordingly, the final constructed map of the environment E will be represented as:

$$\mathsf{E} = \{\mathsf{K}, \mathsf{P}, \mathsf{M}, \mathsf{W}, \mathsf{R}\} \tag{1}$$

where $\mathsf{K} = \{k\}$ F is the set of keyframes and $\mathsf{P} = \{\mathsf{p}\}$ represents the set of feature points $\mathsf{p} = \{\mathsf{x}, \mathsf{v}, \hat{\mathsf{d}}\}$ extracted from the environment with their corresponding 3D positions $\mathsf{x} \in \mathbb{R}^3$, viewing direction $\mathsf{v} \in \mathbb{R}^3$, and descriptor $\hat{\mathsf{d}}$. Additionally, $\mathsf{M} = \{\mathsf{m}\}$ is the set of ArUco markers detected in the environment, in which each marker $\mathsf{m} = \{s, \mathsf{p}, \mathsf{cc}\}$ holds marker size (*i.e.,* length) $s \in \mathbb{R}$, marker pose $\mathsf{p} \in SE(3)$ calculated from M_t to G_t, and corner coordinates $\mathsf{cc} = (c_1 \dots c_4) | c_i \in \mathbb{R}^3$ values. The set of walls detected from the environment is represented by $\mathsf{W} = \{\mathsf{w}\}$, in which each wall $\mathsf{w} = \{\mathsf{q}, \mathsf{m}_\mathbf{w}\}$ holds the wall equation $\mathsf{q} \in \mathbb{R}^4$ and the attached markers list $\mathsf{m}_\mathbf{w}$ $\mathsf{M} = (m_1 \dots m_n) | m_i \in \mathbb{N}$ where m_i represents ArUco *marker-id*. Similarly, $\mathsf{R} = \{\mathsf{r}\}$ refers to the set of rooms found in the environment, where each room $\mathsf{r} = \{\mathsf{r}_\mathbf{c}, \mathsf{r}_\mathbf{w}\}$ contains the room center point $\mathsf{r}_\mathbf{c} \in \mathbb{R}^3$ and the wall list $\mathsf{r}_\mathbf{w}$ $\mathsf{W} = (w_1 \dots w_n) | w_i \in \mathbb{N}$ that comprise the room.

B. Semantic Entities

Walls. Each wall plane w_i is extracted in the global coordinate system ${}^G w_i = {}^G n_i \quad {}^G d$ with a normal vector ${}^G n_i = \begin{bmatrix} n_x & n_y & n_z \end{bmatrix}^T$. The vertex node of the wall is factored in the graph as ${}^G w_i = [{}^G , {}^G , {}^G d]$, where ^{M} and ^{M} refer to the azimuth and elevation of the wall in G_t. For each marker ${}^G m_i$ attached to the wall ${}^G w_i$ cost function can be defined as:

$$c_{w_i}({}^G w_i, {}^G m_i) = k[{}^M_{w_i m_i}, {}^M_{w_i m_i}, {}^M d_{w_i}]^T k^2_{w_i} \tag{2}$$

where ${}^M_{w_i m_i}$ difference between the azimuth angle of the wall w_i and its marker m_i converted to its marker frame M_i, ${}^M_{w_i m_i}$ is the difference in the elevation angles, while ${}^M d_{w_i}$ being the perpendicular distance between the wall and the marker, which should be zero the given marker-wall pair.

Rooms. Since perceiving a room can be difficult due to various configurations and structures, the proposed approach employs the data encoded in ArUco markers attached to the room's walls to detect the mentioned semantic entity. In this regard, a dictionary containing the rooms in the environment and the fiducial markers attached to their walls are provided to feed the framework. Note that the only information encoded in the dictionary is the *marker-ids* corresponding to a room, and no additional pose information is required to be encoded. Hence, two room types titled *"two-wall room"* and *"four-wall room"* have been considered in this work:

Two-wall Rooms (Corridors): In this case, only two parallel walls of a room are labeled with fiducial markers. This scenario is proper for detecting corridors or rooms with undetectable/unreachable walls in the scene. Consequently, a room ${}^G \mathsf{r}_x = [{}^G \mathsf{w}_{x_{a_1}}, {}^G \mathsf{w}_{x_{b_1}}]$ contains x-wall planes parallel to the x-axis while ${}^G \mathsf{r}_y = [{}^G \mathsf{w}_{y_{a_1}}, {}^G \mathsf{w}_{y_{b_1}}]$ contains y-wall planes parallel planes to y-axis.

To compute the center of a two-wall room ${}^G \mathsf{r}_{x_i}$, the two x-wall plane equations are utilized along with the center point ${}^G \mathsf{c}_i$ of the marker m_i as follows:

$$
\begin{aligned}
{}^{G}\mathbf{k}_{x_i} &= \tfrac{1}{2}\,\big|{}^{G}d_{x_{a_1}}\big|\;{}^{G}\mathbf{n}_{x_{a_1}} \quad \big|{}^{G}d_{x_{b_1}}\big|\;{}^{G}\mathbf{n}_{x_{b_1}} \;+\big|{}^{G}d_{x_{b_1}}\big|\;{}^{G}\mathbf{n}_{x_{b_1}} \\
{}^{G}\quad_{x_i} &= {}^{G}\hat{\mathbf{k}}_{x_i} + {}^{G}\mathbf{c}_i \quad [\,{}^{G}\mathbf{c}_i \quad {}^{G}\hat{\mathbf{k}}_{x_i}]\quad {}^{G}\hat{\mathbf{k}}_{x_i}
\end{aligned} \tag{3}
$$

where ${}^{G}\quad_{x_i}$ is the center point of the two-wall room ${}^{G}\mathbf{r}_{x_i}$ and ${}^{G}\hat{\mathbf{k}}_{x_i}$ is acquired from ${}^{G}\hat{\mathbf{k}}_{x_i} = {}^{G}\mathbf{k}_{x_i}{=}k^{G}\mathbf{k}_{x_i}k$. The center point ${}^{G}\mathbf{c}_i$ of the marker is obtained using the marker pose in frame G.

A two-wall room node is initialized using the room center, and the cost function to minimize the two-wall room node and their corresponding wall planes are defined below:

$$
\begin{aligned}
&c_{r_{x_i}}({}^{G}\mathbf{r}_{x_i};\,{}^{G}\mathbf{w}_{x_{a_1}};{}^{G}\mathbf{w}_{x_{b_1}};{}^{G}\mathbf{c}_i\,) \\
&\quad = \sum_{t=1;i=1}^{T\times K} k^{G}\wedge_{x_i} \quad f({}^{G}\mathbf{w}_{x_{a_1}};{}^{G}\mathbf{w}_{x_{b_1}};{}^{G}\mathbf{c}_i)k^2{}_{\epsilon_{i;t}}
\end{aligned} \tag{4}
$$

where $f({}^{G}\mathbf{w}_{x_{a_1}};{}^{G}\mathbf{w}_{x_{b_1}};{}^{G}\mathbf{c}_i)$ maps the wall planes to the center point of the room using Eq. 3.

Four-wall Rooms: A four-wall room contains four wall planes as ${}^{G}\mathbf{r}_i = [{}^{G}\mathbf{w}_{x_{a_1}}\,{}^{G}\mathbf{w}_{x_{b_1}}\,{}^{G}\mathbf{w}_{y_{a_1}}\,{}^{G}\mathbf{w}_{y_{b_1}}]$ forming the room. The center point of this variant of rooms can be computed using the equation below:

$$
\begin{aligned}
{}^{G}\mathbf{q}_{x_i} &= \tfrac{1}{2}\,\big|{}^{G}d_{x_{a_1}}\big|\;{}^{G}\mathbf{n}_{x_{a_1}} \quad \big|{}^{G}d_{x_{b_1}}\big|\;{}^{G}\mathbf{n}_{x_{b_1}} \;+\big|{}^{G}d_{x_{b_1}}\big|\;{}^{G}\mathbf{n}_{x_{b_1}} \\
{}^{G}\mathbf{q}_{y_i} &= \tfrac{1}{2}\,\big|{}^{G}d_{y_{a_1}}\big|\;{}^{G}\mathbf{n}_{y_{a_1}} \quad \big|{}^{G}d_{y_{b_1}}\big|\;{}^{G}\mathbf{n}_{y_{b_1}} \;+\big|{}^{G}d_{y_{b_1}}\big|\;{}^{G}\mathbf{n}_{y_{b_1}} \\
{}^{G}\quad_{i} &= {}^{G}q_{x_i} + {}^{G}q_{y_i}
\end{aligned} \tag{5}
$$

where ${}^{G}\quad_{i}$ is the center point of the four-wall room ${}^{G}\mathbf{r}_i$. It should also be noted that Eq. 5 holds true when $|d_{x_1}| > |d_{x_2}|$. The cost function to minimize four-wall room nodes and their corresponding wall plane set is similar to a two-wall room but with minor differences:

$$
\begin{aligned}
&c\,({}^{G}\quad;\,{}^{G}\mathbf{w}_{x_{a_i}};{}^{G}\mathbf{w}_{x_{b_i}};{}^{G}\mathbf{w}_{y_{a_i}};{}^{G}\mathbf{w}_{y_{b_i}}\,) \\
&= \sum_{t=1;i=1}^{T\times S} k^{G}\wedge_{i} \quad f({}^{G}\mathbf{w}_{x_{a_i}};{}^{G}\mathbf{w}_{x_{b_i}};{}^{G}\mathbf{w}_{y_{a_i}};{}^{G}\mathbf{w}_{y_{b_i}})k^2{}_{\sim_{i;t}}
\end{aligned} \tag{6}
$$

where $f({}^{G}\mathbf{w}_{x_{a_i}};{}^{G}\mathbf{w}_{x_{b_i}};{}^{G}\mathbf{w}_{y_{a_i}};{}^{G}\mathbf{w}_{y_{b_i}})$ maps the four estimated wall planes to the center point of the four-wall room using Eq. 5.

C. Final Graph

Fig. 2 depicts the structure of the final semantic graph produced by the proposed approach. Accordingly, the keyframes extracted by the system are the primary sources of information that contain both visual feature points with their corresponding 3D coordinates and visited ArUco markers in the scene. The topmost level of the graph retains rooms detected in the environment using the *marker-ids* and the walls that hold those markers with constraints obtained following Eq. 4 and Eq. 6 for different rooms.

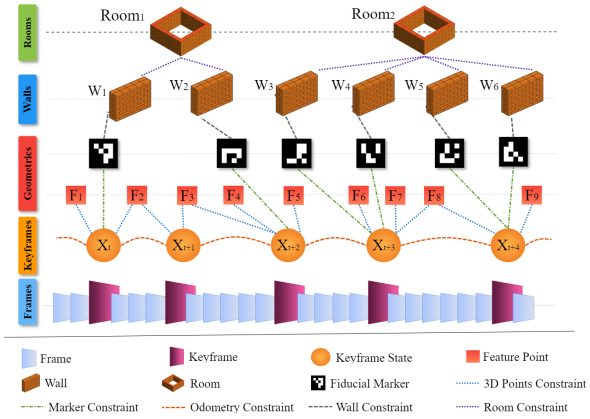

Fig. 2: The graph representation of the hierarchical architecture of the proposed approach.

IV. Evaluation

For evaluation, various real-world scenario tests were performed using the proposed method, UcoSLAM [5] as the baseline methodology, and *S-Graph+* [9] as a Light Detection And Ranging (LiDAR)-based approach for providing ground truth measurements.

A. Evaluation Setup

In order to evaluate the performance of the proposed approach in real-world circumstances, we mounted a *Intel® RealSense™ Depth Camera D435* as the monocular sensor on a *Boston Dynamics Spot®* robot and collected data from an indoor environment. The robot functioned in different office zones of two different university buildings with various corridor and room setups, where the walls were labeled with printed $17cm \times 17cm$ ArUco markers. *Marker-ids* of the ArUco markers placed in the environment, along with the room unique labels, were also stored in a database file and fed to the system.

B. Experimental Results

In order to demonstrate the accuracy of the proposed method compared to its baseline and ground truth, Absolute Trajectory Error (ATE) measurements have been employed in this paper. Accordingly, the Root Mean Square Deviation (RMSE) and Standard Deviation (STD) values of the proposed and baseline approaches were compared to the ground truth, and the approach with less value is assumed to perform more accurately.

According to the evaluation results presented in Table I, the proposed approach works better than its baseline in most of the cases. The main reason for such improvement is the ability of the proposed method to add new constraints to the map and employ the association of semantic entities to enhance the reconstruction of the final map.

The above-mentioned impact is more obvious in cases such as *Seq-04* where the robot starts in a corridor, enters from one of the two doors of a room and exits from the other door to continue in the same corridor, and as a result, no loop closure using keypoints and markers is performed. While our