World Electric Vehicle Journal

MDPI

*Article*

# Fusing Direct and Indirect Visual Odometry for SLAM: An ICM-Based Framework

**Jeremias Gaia [1,*], Javier Gimenez [1], Eugenio Orosco [1], Francisco Rossomando [1], Carlos Soria [1] and Fernando Ulloa-Vásquez [2]**

[1] Instituto de Automática, Universidad Nacional de San Juan (UNSJ)—Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Av. San Martín (Oeste) 1109, San Juan J5400ARL, Argentina; jgimenez@inaut.unsj.edu.ar (J.G.); eorosco@inaut.unsj.edu.ar (E.O.); frosoma@inaut.unsj.edu.ar (F.R.); csoria@inaut.unsj.edu.ar (C.S.)

[2] Programa de Investigación Radio Digital, PIRD, Facultad de Ingeniería, Universidad Tecnológica Metropolitana, Santiago de Chile 7750000, Chile; fulloa@utem.cl

* Correspondence: jgaia@inaut.unsj.edu.ar

## Abstract

The loss of localization in robots navigating GNSS-denied environments poses a critical challenge that can compromise mission success and safe operation. This article presents a method that fuses visual odometry outputs from both direct and feature-based (indirect) methods using Iterated Conditional Modes (ICMs), an efficient iterative optimization algorithm that maximizes the posterior probability in Markov random fields, combined with uncertainty-aware gain adjustment to perform pose estimation and mapping. The proposed method enhances the performance of visual localization and mapping algorithms in low-texture or visually degraded scenarios. The method was validated using the TUM RGB-D benchmark dataset and through real-world tests in both indoor and outdoor environments. Outdoor experiments were conducted on an electric vehicle, where the method maintained stable tracking. These initial results suggest that the technique could be transferable to electric vehicle platforms and applicable in a variety of real-world conditions.

**Keywords:** sensor fusion; visual odometry; Simultaneous Localization and Mapping (SLAM); localization; Iterated Conditional Modes (ICMs)

check for updates

Academic Editor: Vladimir Katic

Received: 22 July 2025
Revised: 20 August 2025
Accepted: 30 August 2025
Published: 10 September 2025

**Citation:** Gaia, J.; Gimenez, J.; Orosco, E.; Rossomando, F.; Soria, C.; Ulloa-Vásquez, F. Fusing Direct and Indirect Visual Odometry for SLAM: An ICM-Based Framework. *World Electr. Veh. J.* **2025**, *16*, 510. https://doi.org/10.3390/wevj16090510

**Copyright:** © 2025 by the authors. Published by MDPI on behalf of the World Electric Vehicle Association. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Simultaneous localization and mapping enables robotic systems to concurrently construct a representation of their environment and determine their own position within that representation [1]. For mobile robots, whether they move autonomously or with the assistance of an operator, the integration of SLAM systems in GNSS-denied environments is imperative [2]. Typically, these systems rely on the fusion of data from diverse sensor suites, such as monocular and stereoscopic vision systems, LiDAR, and inertial sensors [3,4].

With the rise of electric vehicles (EVs), visual-based localization has gained attention as a cost-effective alternative to LiDAR-centric systems. A recent review highlights the growing interest in multi-sensor fusion SLAM approaches for intelligent EV navigation [5]. Moreover, recent studies on visual perception systems in electric vehicles emphasize the importance of robust feature tracking and illumination-invariant algorithms to ensure localization reliability under dynamic and visually degraded conditions [6].

When cameras are the only sensor in the system, visual SLAM (V-SLAM) is performed. V-SLAM techniques are based on visual odometry (VO), which can be defined as the method

for estimating an agent's egomotion (human, robot, vehicle, etc.) based only on the input of a single or several cameras [7–9]. Within this domain, two primary image processing paradigms can be distinguished: direct methodologies, which use pixel brightness and position for motion estimation, and feature-based methodologies, which construct a scene representation prior to motion estimation.

Direct methods select a subset of pixels through a pixel selection strategy and apply a direct image alignment process to estimate camera motion. The influential contributions of Engel et al. [10–12] are pivotal for understanding the direct visual odometry. In [12], the authors proposed estimating camera motion by performing the minimization of a photometric error function derived from the observed pixel intensities. Engel's ideas inspired a significant amount of recent literature [13–16].

Feature-based visual odometry pipeline, on the other hand, starts by extracting a set of features from the input frame, followed by a feature matching procedure with the previous frame. The resulting correspondences are then used to estimate camera motion. The cornerstone of current feature-based methods [17–21] is ORB-SLAM [22]. This system performs the aforementioned steps based on the extraction and matching of a set of ORB descriptors from the input frame. After an initial guess of the agent's pose is obtained, this system refines the pose by a process named motion-only Bundle Adjustment.

While both approaches have proven effective, each comes with its limitations. Direct methods assume photometric consistency (i.e., pixel intensities remain constant across frames), a condition that can be easily violated by lighting changes, leading to errors in the motion estimation. Feature-based methods, on the other hand, suffer in low-texture environments where feature extraction and matching become unreliable, often resulting in tracking loss. Despite these limitations, the two methods are complementary. In low gradient scenarios, direct methods can still recover a good pose estimate, while feature-based methods are resilient to brightness variations when reliable features are present.

This work proposes a modular fusion architecture that combines the outputs of direct and feature-based visual odometry within a single framework. By exploiting their complementary characteristics, the system mitigates the risk of full tracking loss and promotes more reliable pose estimation.

Since the beginning of the 21st century, the SLAM problem has been extensively studied using graph-based models. Folkesson and Christensen introduced the GraphSLAM system which finds the best robot trajectory using a nonlinear optimization technique [23]. Graph theory allows one to intuitively illustrate the states of a system with their interdependencies. In a SLAM context, nodes represent the robot's poses at different time steps, and edges encode spatial constraints between these poses. These constraints arise from sensor observations or the robot's motion. Once this graph is established, the map is obtained by estimating the spatial arrangement of the nodes that best aligns with the constraints defined by the edges [24].

Although graph-based methods offer a strong framework, full pose graph optimization heavily relies on a loop closure event. Without this crucial constraint, the optimization problem lacks sufficient information, leading to reduced global accuracy in the estimated map and trajectory [25].

Markov random fields (MRFs) can be used to solve the SLAM problem if the conditional probabilities of the model are consistent with the topological structure of the graph [26]. The potentiality of working in the MRF context has been reported in [27], where the authors proposed a fully Bayesian way to solve the simultaneous localization and spatial prediction (SLAP) problem by using a Gaussian Markov random field (GMRF) model with satisfactory results. In [28], GMRFs are integrated for modeling information distribu-

tion. The study presents a computationally efficient strategy for autonomous exploration in 3D environments.

In a previous study [29], Gimenez et al. proposed a general method for localization and mapping in the MRF context. This method employs the Iterated Conditional Modes (ICMs) algorithm, which systematically finds a configuration of states that locally maximizes the posterior probability, given a set of observations, within a Markovian model. The approach was evaluated using a differential drive mobile robot equipped with a laser range finder. The robot traversed different structured environments with a finite set of known landmarks. In a subsequent study [30], the mapping is based on a point cloud generated from successive environmental observations, with each location assigned an occupancy probability using 2D kernel density estimation (KDE). To manage point cloud growth and adapt to dynamic environments, the system incorporates a novel recursive subsampling method that also helps in gradually discarding moving objects. This approach facilitates online point probability calculations and bypasses the complex data association problem inherent in other map representations. In both use cases, the system successfully recovered the robot's trajectory and produced accurate map estimates. In this paper, we present an enhanced version of the framework described in [29]. The proposed framework is designed to incorporate diverse observation models (regardless of the sensor type and data characteristics), odometry sources, and any map representation. Our contributions can be summarized as follows:

- We propose a novel, modular approach for the unified fusion of direct and feature-based odometry. This integration improves pose estimation reliability and reduces tracking failure, particularly in challenging conditions.
- Novel potential functions are introduced to enable 6-DoF pose estimation using ICM.
- Dynamically adjusted gain values are integrated into the ICM algorithm to enhance robustness against tracking loss in visually degraded or low-texture environments.

The remainder of the paper is organized as follows. Section 2 provides an overview of the system and the theoretical foundations of the ICM method, followed by a description of the proposed iterative fusion process and the adaptable gain value scheme. Section 3 presents the results obtained from both benchmark datasets and real-world experiments. The findings are discussed in Section 4, and the conclusions are drawn in Section 5.

## 2. Materials and Methods

### *2.1. System Overview*

Figure 1 illustrates the modular structure of the proposed system. Input RGB-D frames are simultaneously processed by a feature-based VO module and a direct VO module, each producing an independent estimate of the agent's pose. At each time step, the direct VO module also generates a set of 2000 observations, which are subsequently transformed into a point cloud. These observations are fed into a dedicated map handler that builds and maintains both a local map and a global map. The local map supports internal computations, while the global map provides a visual representation of the reconstructed environment.

The ICM module receives the odometry measurements, the local map, and the observations to yield a refined estimate of the camera's pose $x_t$. Inside this module, each one of the inputs is represented as a potential function. As will be described in Section 2.4.1, it is through these potential functions that the fusion of measurements from direct and feature-based odometry is achieved.

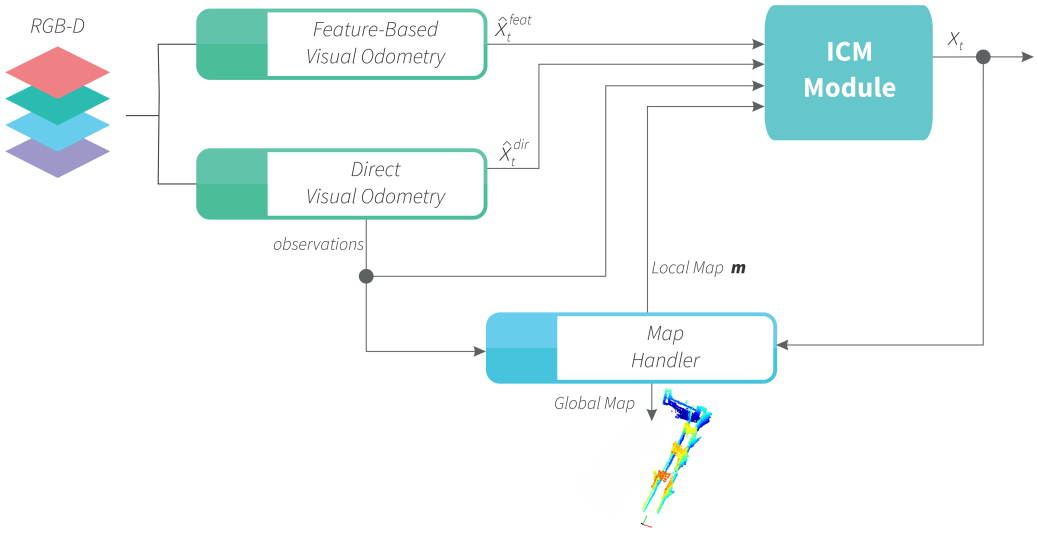



**Figure 1.** System overview as a block diagram. The inputs $\hat{x}_t^{feat}$ and $\hat{x}_t^{dir}$ are interpreted as odometry measurements by the ICM module, which additionally receives observation data and the local map **m** for pose estimation.

The 6-DoF pose vector of the camera at each time is defined as $\mathbf{x_t} = [x, y, z, \phi, \theta, \psi]$, with $x$, $y$ and $z$ being the translational part of the vector, and $\phi$, $\theta$, and $\psi$ being the associated angles corresponding to rotations in the x, $y$ and $z$ axes, respectively. To enable a direct comparison with the ground truth data available in state-of-the-art datasets, the output pose of the proposed system is expressed in the Lie group $SE(3)$ as

$$\xi = \begin{bmatrix} \Gamma & \tau \\ \mathbf{0} & 1 \end{bmatrix} \tag{1}$$

where $\tau = [x, y, z]^T$ of the $\mathbf{x_t}$ vector and the rotation matrix $R$, also denoted as $R(\phi, \theta, \psi)$, is defined as the concatenation of the rotations around each coordinate axis

$$\Gamma = \Gamma(\phi, \theta, \psi) = \Gamma_z(\psi)\Gamma_y(\theta)\Gamma_x(\phi) \tag{2}$$

where

$$\Gamma_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}, \Gamma_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}, \Gamma_z(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The design of the odometry modules is inspired by state-of-the-art systems such as ORB-SLAM and DSO. Nevertheless, such systems require C++-based libraries to perform pose estimation and optimization, which posed compatibility challenges for this study's Python 3.8-based framework. Therefore, a custom implementation was developed to maintain the full control over the pipeline and ensure portability.

### *2.2. Theoretical Background on Iterated Conditional Modes*

Given a finite set of nodes $\mathcal{V}$ in a graph, a random field (RF) is an array of random variables $\mathbf{X} = \{X_v : vs. \in \mathcal{V}\}$. The subset of random variables corresponding to a node subset $A \subseteq \mathcal{V}$ is denoted as $X_A = \{X_a : a \in A\}$, while the set of all variables except $X_v$ is written as $X_{-v} = \{X_a : a \in \mathcal{V}, a \neq vs.\}$. Two nodes are said to be neighbors if they are connected by an edge in the graph, and the set of all neighbors of a node $v$ is denoted by $\partial v$. A clique $C \subseteq \mathcal{V}$ is a subset of nodes in which every pair of nodes is mutually connected. The set of all cliques in the graph is denoted by $\mathcal{C}$.

Let $\Omega$ represent the set of all possible states of the random field $\mathbf{X}$, and let $P$ be the probability defined over $\Omega$ such that $P(\omega)$ denotes the probability of the RF $\mathbf{X}$, assuming the state $\omega$. The conditional probability that $X_v$ takes a value $\omega_v$ given that $X_A$ takes value $\omega_A$ is denoted by $P(\omega_v|\omega_A)$. Then, a random field $\mathbf{X}$ defined over a graph with associated probability $P$ is a Markov random field (MRF) if

$$P(\omega_v|\omega_{-v}) = P(\omega_v|\omega_{\partial v}) \quad \forall vs. \in \mathcal{V} \tag{3}$$

That is, the probability of observing each $\omega_v$, given all other states, can be computed solely from the values in its neighborhood.

Now, consider a family of potentials $\{\Phi_C : C \in \mathcal{C}\}$, where each $\Phi_C$ is a function that assigns a real-valued confidence measure to each local configuration $\omega_C$, such that

$$\Phi_C(\omega_C) > \Phi_C(\omega'_C) \tag{4}$$

when configuration $\omega_C$ is more likely than $\omega'_C$.

Then, assuming

$$P(\omega) \propto exp\left\{ \sum_{C \in \mathcal{C}} \Phi_C(\omega_C) \right\} \tag{5}$$

then it can be shown that

$$P(\omega_v|\omega_{\partial v}) = P(\omega_v|\omega_{-v}) \propto exp\left\{ \sum_{C \in \mathcal{C}: vs. \in C} \Phi_C(\omega_C) \right\} \tag{6}$$

Finding the global maximum of the joint probability defined in Equation (5) requires an appropriate optimization method. Simulated annealing (SA) is a well-known iterative approach capable of addressing this task. Although SA guarantees theoretical convergence to the global optimum, its application is often impractical due to its high computational cost. Given these conditions, the Iterated Conditional Modes (ICM) algorithm rises as a practical alternative. This method iteratively optimizes the conditional probabilities in Equation (6) at a lower computational cost. While ICM's resulting configuration typically converges to the local maximum of the Equation (5) closest to the initial state, it can yield highly satisfactory results when the initialization is appropriately chosen [31].

### *2.3. Potential Functions Definition*

In order to solve for the most likely configuration of the system at each timestep $t$, it is necessary to define potential functions $\Phi$. Each $\Phi$ must assign a real-valued confidence measure to each local configuration, and fulfill Equation (4).

The flexibility of this proposal lies in the fact that it is possible to define a $\Phi$ function for each information source within the system, regardless of its nature. The most important design requirement is that these functions must penalize incorrect configurations while rewarding correct ones.

As seen in Equations (5) and (6), the objective function is expressed as the sum of all system potentials. This formulation allows an additional layer of flexibility, as each potential can include a gain factor $K$. As a result, Equation (5) becomes a weighted sum, where each term can reflect the relative confidence of the user in the corresponding sensor measurement.

Next, we describe the potential functions used for the experimental evaluation of this system. Subscripts on the pose vector indicate the time range considered for estimation; for instance, $x_{t-1:t+1}$ denotes that poses at time steps $t-1$, $t$, and $t+1$ are included in the

calculation. Additional information and definitions referring to other types of sensors can be found in [30].

#### 2.3.1. Odometry Potential

In this article, we define odometry $\hat{x}_t$ as any estimation of the pose $x_t$ obtained from internal sensory inputs. Over time, such estimates are prone to substantial drift due to accumulated error. However, if the relative change between consecutive estimations is considered, then $x_t - x_{t-1}$ can be estimated with $\hat{x}_t - \hat{x}_{t-1}$. A possible way to express this relationship could be

$$\Phi^{\text{odo}}_{\{x_{t-1:t}\}}(x_{t-1:t}) = -K_{\text{odo}}||s(x_{t-1:t}) - s(\hat{x}_{t-1:t})||^2 \tag{7}$$

where $s(x_{t-1:t})$ is defined as

$$s(x_{t-1:t}) = R_c(x_{t-1;\phi,\theta,\psi})(x_{t;x,y,z} - x_{t-1;x,y,z}) \tag{8}$$

and $R_c(x_{t-1;\phi,\theta,\psi}) = \Gamma^T$ is a matrix designed to cancel the rotation of the odometry vector, with $\Gamma$ defined in Equation (2).

#### 2.3.2. Continuity Potential

To ensure smooth pose transitions over time, introducing a potential function that penalizes abrupt positional changes may be beneficial.

The expression in Equation (9) enforces the value of $x_t$ to remain close to those of its neighboring nodes in the graph.

$$\Phi^{\text{cont}}_{\{x_{t-2:t}\}}(x_{t-2:t}) = -K_{\text{cont}}||s(x_{t-2:t-1}) - s(x_{t-1:t})||^2 \tag{9}$$

This potential function reduces the system's sensitivity to inaccurate pose estimates caused by poor data association or outliers.

#### 2.3.3. Mapping and Observations Potential

Let us consider the general case of an agent navigating through an unknown environment. Traditional landmark-based SLAM systems depend on the presence and constant re-observation of distinctive landmarks of the map. Without prior knowledge of their locations, these systems can become infeasible in practice, especially in unfamiliar or large-scale settings [32].

Now, suppose that the system is equipped with a sensor array capable of creating an abstraction of the environment (e.g., a point cloud). Additionally, assume it has sufficient memory to store observations acquired at neighboring nodes in the graph. Under these conditions, this memory can be interpreted as a local map $\mathbf{m}$. Each point in the point cloud is an observation of the environment.

At each time step $t$, the system captures a set of observations $z_t = \{z_{t,i} : i = 1, \ldots, n_t\}$, where $n_t$ denotes the number of points observed at that moment. These observations establish a connection between the robot's pose $x_t$ and $\mathbf{m}$. The local map $\mathbf{m}$ is constructed by aggregating observations from several time steps. As a result, the local map contains $N_l$ points, with $N_l > n_t$, since it combines information from multiple frames.

Once the nature of the observations and the map has been established, a potential function can be defined to quantify the correspondence between the set of observations and the map. If the $N_l$ points in the local map are treated as known position landmarks for the current time, then the corresponding observation potential can be defined as

$$\Phi^{\text{obs}}_{\{x_t,\mathbf{m}\}}(x_t,\mathbf{m}) = -K_{\text{obs}}\frac{1}{n_t}\sum_{i=1}^{n_t}\arg\min_{l}||h(x_t,z_{t,i}) - m_l||^2 \tag{10}$$

where $h(x_t, z_{t,i})$ represents the position of a given observation $z_{t,i}$ given $x_t$ and $m_l \in N_l$ is a point of the local map.

In this article, observations take the form of point clouds. Each element of the point cloud is represented as $z_{t,i} = [z_{t,i,x}, z_{t,i,y}, z_{t,i,z}] \in \mathbb{R}^3$. Given $x_t$, the position of $z_{t,i}$ is obtained by

$$h(x_t, z_{t,i}) = \Gamma(t)z_{t,i} + \tau(t) \tag{11}$$

with $\Gamma(t) = \Gamma(x_{t,\phi}, x_{t,\theta}, x_{t,\psi})$ and $\tau(t) = [x_{t,x}, x_{t,y}, x_{t,z}]$

### 2.4. Data Fusion via ICM

In this section, the iterative process for the sensor fusion via an ICM implementation is described. First, a method to initialize the system states is proposed in Section 2.4.1. Next, the optimization process is described in Section 2.4.2.

However, before delving into the mathematical concepts relevant to data fusion through ICM, these ideas are first graphically illustrated using the example shown in Figure 2. The system states are initialized in what is referred to as iteration 0 of the algorithm, displayed on the left-hand side of the figure. To identify the optimal configuration of the states (highlighted in the image with the light-blue triangle), the algorithm draws upon present and past information from the defined potentials. In this case, it integrates information from direct and feature-based odometry, the continuity potential, and the observation potential. For the clarity of presentation, only the measurements from the current time step of each potential are depicted in the image.

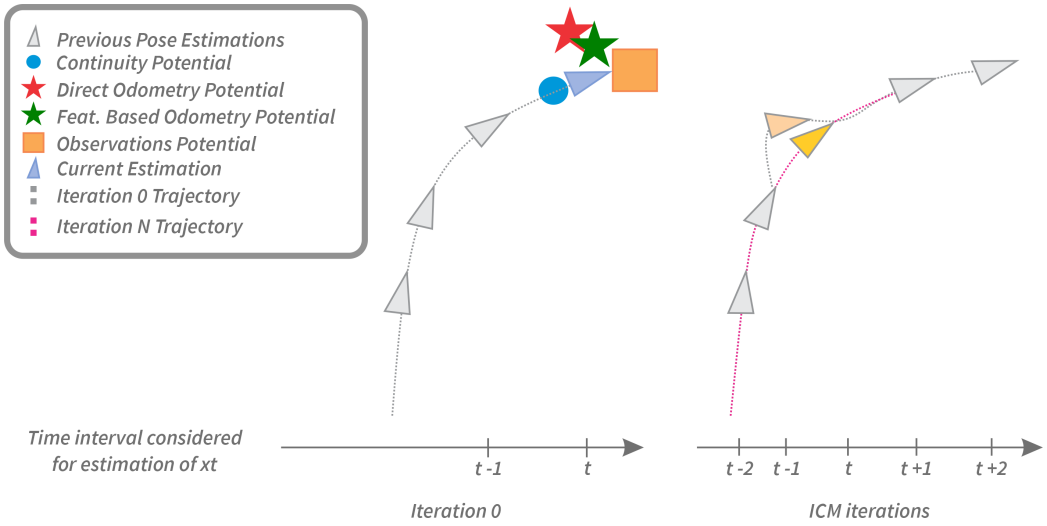



**Figure 2.** Two−phase process of the ICM Algorithm. (**Left**) The initialization phase (iteration 0) determines the current system pose using past and present data. (**Right**) N iterations if the ICM algorithm are performed to refine the trajectory estimates. To update a state at a given time step, this process leverages all available information, including past and future states.

After the exploration concludes, the system halts, or a loop closure occurs, and the ICM algorithm transitions into a second phase, referred to as ICM iterations (right side of Figure 2). Unlike iteration 0, where future information is unknown, iterations N refine the trajectory using the complete set of past, present, and future data from the potential functions. In this way, the pose at a given time (e.g., marked in transparent orange) is optimized into a refined estimate (yellow), improving upon the initial iteration.

#### 2.4.1. Initialization

As stated in Section 2.2, the ICM algorithm converges to a high-quality local maximum if the initial states are appropriately chosen. Thus, the success of the method relies heavily on the quality of the initialization phase (iteration 0 of ICM). This phase proceeds until the system stops, exploration ceases, or a loop closure is detected. Let $T_f$ represent the time at which iteration 0 terminates under any of these conditions.

During the exploration of an unknown environment, the robot only has access to current and past data. We propose an online initialization strategy that leverages this information to obtain an initial estimate of the robot's trajectory $\mathbf{x}^{(0)}$ and the local map $\mathbf{m}^{(0)}$. Algorithm 1 describes how to compute the pose $x_t^{(0)}$ given the available data.

**Algorithm 1** Online ICM initialization.

**Ensure:** $x_0^{(0)}, \mathbf{m}^{(0)}$
**while** $t < T_f$ **do**

$$\begin{aligned} x_t^{(0)} = \arg\min_{\mathbf{x}} \Big\{ &- K_{\text{odo}} ||s(x_{t-1:t}) - s(\hat{x}_{t-1:t})||^2 - \\ &K_{\text{cont}} ||s(x_{t-2:t-1}) - s(x_{t-1:t})||^2 - \\ &- K_{\text{obs}} \frac{1}{n_t} \sum_{i=1}^{n_t} \arg\min_{l} ||h(x_t, z_{t,i}) - m_l||^2 \Big\} \end{aligned} \tag{12}$$

Update local map $\mathbf{m}^{(0)}$ given the estimated $x_t^{(0)}$ and evaluate candidate points for the global map
**end while**

In this implementation, the integration of both odometry sources leads to the following extended form of Equation (12):

$$\begin{aligned} x_t^{(0)} = \arg\min_{\mathbf{x}} \Big\{ &K_{\text{odo}}^{dir} ||s(x_{t-1:t}) - s(\hat{x}_{t-1:t}^{dir})||^2 + K_{\text{odo}}^{feat} ||s(x_{t-1:t}) - s(\hat{x}_{t-1:t}^{feat})||^2 + \\ &K_{\text{cont}} ||s(x_{t-2:t-1}) - s(x_{t-1:t})||^2 + K_{\text{obs}} \frac{1}{n_t} \sum_{i=1}^{n_t} \arg\min_{l} ||h(x_t, z_{t,i}) - m_l||^2 \Big\} \end{aligned} \tag{13}$$

In this way, Equation (13) enables the fusion of the outputs from the odometry systems with each other and with the continuity and observation potentials, so that, during optimization, state configurations inconsistent with the pose graph are penalized while correct configurations are rewarded.

#### 2.4.2. ICM Iterations

Once the system states have been initialized, an optimization procedure is performed using $N$ ICM iterations. This process leverages information from the state currently being updated as well as from past and future data. As a result, Equations (7) and (9) can be modified to account for the future data as follows:

$$\Phi^{\text{odo}}_{\{x_{t-1:t+1}\}}(x_{t-1:t}) = -K_{\text{odo}} \left[ \Phi^{\text{odo}}_{\{x_{t-1:t}\}}(x_{t-1:t}) + \Phi^{\text{odo}}_{\{x_{t:t+1}\}}(x_{t:t+1}) \right] \tag{14}$$

$$\Phi^{\text{cont}}_{\{x_{t-2:t+2}\}}(x_{t-2:t}) = -K_{\text{cont}} \left[ \Phi^{\text{cont}}_{\{x_{t-2:t}\}}(x_{t-2:t}) + \Phi^{\text{cont}}_{\{x_{t-1:t+1}\}}(x_{t-1:t+1}) + \Phi^{\text{cont}}_{\{x_{t:t+2}\}}(x_{t:t+2}) \right] \tag{15}$$

During the optimization process detailed in Algorithm 2, a sweep over the states obtained in the initialization is performed. These states are arbitrarily sorted according to

a visit scheme (e.g., $x_0$, $x_1$,..., $x_{T_f}$). In this step, each node is visited according to the visit scheme, jointly optimizing (10), (14), and (15) to find the most suitable configuration of the system. This process is repeated during $N$ ICM iterations.

**Algorithm 2** ICM iterations.

**Ensure:** $\mathbf{x}^{(0)}, \mathbf{m}^{(0)}$, and a number of ICM iterations to perform **N**.

Generate a visit scheme, i.e., choose in what order the model states ($x_1, x_2$..., $x_{T_F}$) will be updated.

**for** n < $N$ **do**

**for** each $x_t$ in the visit scheme **do**

$$x_t^{(n)} = \arg\min_{\mathbf{x}} \Big\{ -K_{\text{odo}}\big[||s(x_{t-1:t}^{(n)}) - s(\hat{x}_{t-1:t})||^2 + ||s(x_{t:t+1}^{(n-1)} - s(\hat{x}_{t:t+1})||^2\big] + -K_{\text{cont}}\big[||s(x_{t-2:t-1^{(n)}}^{(n)}) - s(x_{t-1:t}^{(n)})||^2 + ||s(x_{t-1:t}^{(n)}) - s(x_{t:t+1}^{(n-1)})||^2 + ||s(x_{t:t+1}) - s(x_{t+1:t+2}^{(n-1)})||^2\big] + -K_{\text{obs}}\frac{1}{n_t}\sum_{i=1}^{n_t} \arg\min_{l} ||h(x_t, z_{t,i}) - m_l^{(n)}||^2 \Big\}$$

**end for**

**end for**

Update global map given the optimized trajectory.

Note that the mathematical expression for the observations' potential does not change. However, given that posterior information is available, the local map used in this function will be built in each iteration with observations from $x_{t-2}$, $x_{t-1}$, $x_t$, $x_{t+1}$, and $x_{t+2}$. The number $N$ of ICM iterations can be as large as needed, and depends on the quality of the initialization. Additionally, a termination criterion can be added to stop iterating.

### *2.5. Adaptable Gain Values*

To significantly reduce the likelihood of tracking failure, the system needs to dynamically adapt to varying visual conditions. One way to achieve this is to change the effect of the odometry measurements in the final result.

In [10], the authors considered the inverse of the Hessian from the last iteration in the pose estimation algorithm as an estimate of the covariance of the estimation. Under this assumption, we propose to extract an uncertainty measure for the pose estimation provided by the visual odometry modules by

$$\sigma_\xi = \left(\prod_{i=1}^{n} \sqrt{v_i}\right)^{1/n} \tag{16}$$

where $\xi$ refers to the pose representation (1), and $v_i$ represents the diagonal elements of the Hessian inverse matrix. We consider that the aforementioned Hessian matrix is available to the system.

Odometry gains are expected to behave inversely with respect to the measurement uncertainty (16). Higher uncertainty should yield lower gains, while reliable measurements should result in higher gains. To model this relationship, a gain adjustment function is defined as

$$K_{\text{odo}} = e^{\alpha \cdot (\sigma_\xi)^\beta} \tag{17}$$

The parameters $\alpha$ and $\beta$ determine how pose uncertainty affects the weighting factor. Specifically, $\alpha$ regulates the sensitivity of the weight with respect to the uncertainty: when $\alpha$ is negative, higher values of $\Sigma_{\xi}$ lead to a lower value of $K_{\text{odo}}$. Meanwhile, $\beta$ defines the shape of this dependency: a value of $\beta = 1$ results in a linear relationship; values between 0 and 1 attenuate the effect of $\Sigma_{\xi}$; and values greater than 1 amplify it. Combined, $\alpha$ and $\beta$ offer a flexible mechanism for adjusting the influence of uncertainty on the confidence in pose estimates. Based on laboratory experiments, the optimal values were found to be $\alpha = -2$ and $\beta = 0.3$. In Section 3.1, we demonstrate how these values contribute to maintaining tracking performance under challenging scenarios.

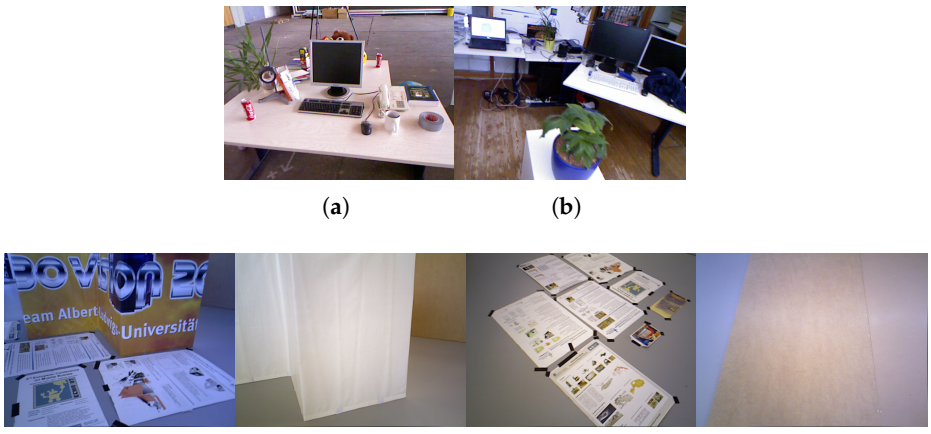
## 3. Results

In this section, we report the results obtained by testing our proposal on the TUM RGB-D benchmark dataset as well as through field tests conducted at different locations on the campus of the Facultad de Ingeniería of the Universidad Nacional de San Juan (FI-UNSJ) and the Instituto de Automática (INAUT) facilities. These sequences provide a basis for evaluating the system's performance across structured, semi-structured, and unstructured environments, and under both indoor and outdoor lighting conditions.

### *3.1. TUM Dataset*

The TUM RGB-D Dataset [33] is an extensively used benchmark in the field of computer vision, particularly for the evaluation and comparison of algorithms related to simultaneous localization and mapping (SLAM) and visual odometry. It was developed by the Computer Vision Group at the Technical University of Munich (TUM), a leading institution in robotics and computer vision research. This dataset comprises synchronized sequences of RGB (color) and depth images captured using a Microsoft Kinect sensor. Additionally, ground truth camera trajectories recorded using a high-precision motion capture system are provided.

Our system was evaluated on six different sequences: *fr1/plant*, *fr2/desk*, *fr3/structure-texture*, *fr3/structure-no texture*, *fr3/no structure-texture*, and *fr3/no structure-no texture*. All these sequences were recorded with a handheld RGB-D Kinect sensor. Figure 3 shows the samples from the different sequences. Note that, in the second row, the conditions for visual pose estimation degrade progressively, ranging from the most favorable (Figure 3c, rich in texture, and structure) to the most challenging scenario (Figure 3f, lacking both structure and texture).



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

**Figure 3.** Samples of the color frames of the TUM RGB-D dataset for the following sequences: (**a**) *fr2/desk*. (**b**) *fr1/plant*. (**c**) *fr3/structure-texture*. (**d**) *fr3/structure-no texture*. (**e**) *fr3/no structure-texture*. (**f**) *fr3/no structure-no texture*.

The metric used for the evaluation is the absolute trajectory error (ATE). It measures the global consistency of the estimated trajectory by computing the absolute distances between points in the estimated and ground-truth trajectories. It is a global measurement that requires both trajectories to be aligned in the same coordinate frame beforehand. One way to achieve this goal is to use the Horn method [34], which finds the rigid-body transformation $\mathbf{P}$ that maps the estimated trajectory $\mathbf{S}_{1:n}$ onto the ground truth trajectory $\mathbf{Q}_{1:n}$.

Given a rigid body transformation between the estimated trajectory and the ground truth data, the ATE at time step $i$ can be computed as

$$ATE_i = \mathbf{Q}_i^{-1}\mathbf{S}\mathbf{P}_i \tag{18}$$

then, the root mean squared error (RMSE) value of (18) is calculated using only the translational component.

$$ATE_{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\|trans(\mathbf{ATE}_i)\|^2} \tag{19}$$

Table 1 presents the absolute trajectory error (ATE) metrics for each of the sequences. The metrics were obtained using the automated evaluation tool provided with the dataset. The following analysis of Table 1 is supported by images depicting the trajectories generated throughout the experimental evaluations.

**Table 1.** Results for the TUM RGB-D dataset.

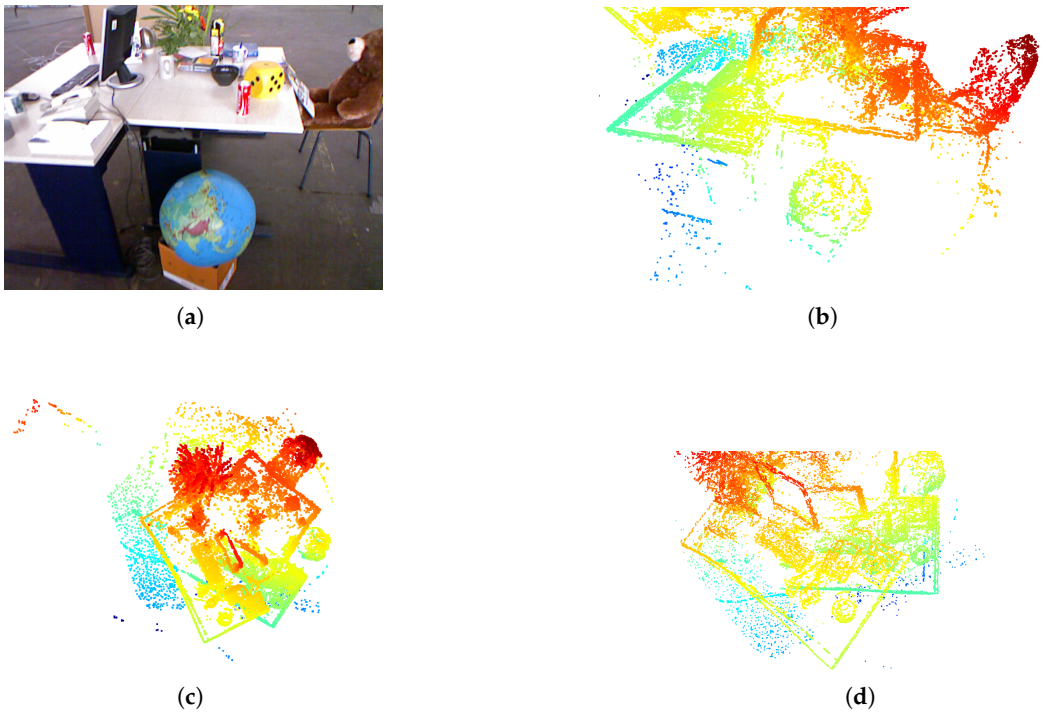
| Sequence | ATE RMSE [m] | ATE Mean [m] | ATE Median [m] | ATE Std [m] | ATE min [m] | ATE max [m] |
|---|---|---|---|---|---|---|
| *fr1/plant* | 0.5567 | 0.4989 | 0.4895 | 0.2470 | 0.0185 | 1.0784 |
| *fr2/desk* | 0.4344 | 0.3900 | 0.3743 | 0.1914 | 0.0321 | 0.7947 |
| *fr3/s-t* | 0.0296 | 0.0275 | 0.0266 | 0.0108 | 0.0035 | 0.0619 |
| *fr3/s-Nt* | 0.1933 | 0.1795 | 0.1821 | 0.0715 | 0.0692 | 0.4366 |
| *fr3/Ns-t* | 0.0574 | 0.0529 | 0.0486 | 0.0222 | 0.0041 | 0.1155 |
| *fr3/Ns-Nt* | 0.3361 | 0.3121 | 0.2990 | 0.1246 | 0.1115 | 0.5243 |

Abbreviations: s-t: Structure–texture, Ns-t: No structure–texture, s-Nt: structure–no texture, Ns-Nt: No structure–no texture.

For sequences such as *fr1/plant* and *fr2/desk*, which contain rich visual features but involve severe perspective changes that challenge visual odometry systems, our proposal presents ATE values around 0.55 for *fr1/plant* and 0.43 for *fr2/desk*.

Figure 4 shows the different angles of the scene reconstruction for this sequence. Specifically, Figure 4c,d illustrate how the pose drift affects reconstruction. The yellow dots at the bottom-left corner of the desktop in Figure 4d represent the same region as the light blue/green dots on the left side of the image.

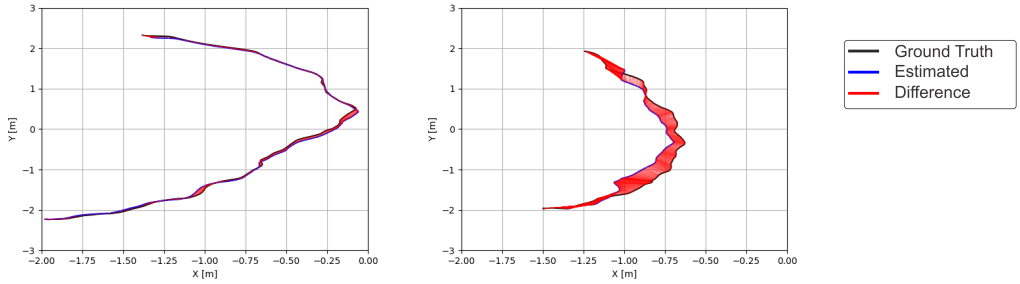
The *structure–texture* family of sequences presents increasingly difficult visual challenges for the pose estimation system. Similarly to the *fr1/plant* and *fr2/desk*, the *fr3/structure–texture* sequence provides images with rich structure and texture. As shown in Table 1, the system achieves its best results in this sequence, with an ATE RMSE of just 0.0296 m in the initial iteration and 0.0365 in the final iteration. This confirms the method's capacity to provide highly accurate estimates when rich structure and texture are available. Figure 5a provides a visual comparison between the estimated trajectory and the ground truth.



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

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

**Figure 4.** Reconstruction details for the *fr2/desk* sequence. (**a**) Reference RGB frame. (**b**) Reconstruction taken from a similar point of view. (**c**) Top view of the final global map for the sequence. (**d**) Detail showing the drift in the reconstruction.




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

**Figure 5.** Pose estimation results for the *structure–texture* sequences. (**a**) *fr3/structure–texture*, (**b**) *fr3/structure–no texture*.

In the *fr3/no structure–texture* sequence, a handheld camera moves along an approximately straight path through an area with posters placed on the floor. Due to the presence of these posters, the sequence contains a high level of texture but no structural information. During this evaluation, the system successfully recovers the camera trajectory, achieving an ATE RMSE of 0.0574 m.

In texture-less sequences such as *fr3/structure–no texture* and *fr3/no structure–no texture*, the impact of the proposed adaptable odometry gain values becomes more apparent. For instance, in the case of the *fr3/structure–no texture* sequence (Figure 5b), the feature-based module can detect a few keypoints from image structure. However, the lack of texture leads to incorrect feature matching (i.e., wrong data association), resulting in poor pose estimation. As a consequence, the system assigns high uncertainty to the output of this module, which translates into a reduced gain. Then, its influence on the final optimization decreases, causing the system to rely more on direct odometry. This behavior of the system is visualized in Figure 6a), where the output of the two odometry systems and the output of the ICM module for the *fr3/structure–no texture* sequence is shown. It can be seen how the

performance of the feature-based module decreases, and the ICM module decides to follow the direct odometry output. Table 1 reports an ATE RMSE of 0.19 m for this sequence, which can be visualized in Figure 6c.

A closer examination of the trajectories shown in Figure 5 reveals the impact of removing texture from the scene. It can be observed that, for a similar trajectory, the feature-based VO performs poorly in low-texture conditions. However, the adaptive gain scheme prevents tracking loss, at the cost of increased pose estimation error.

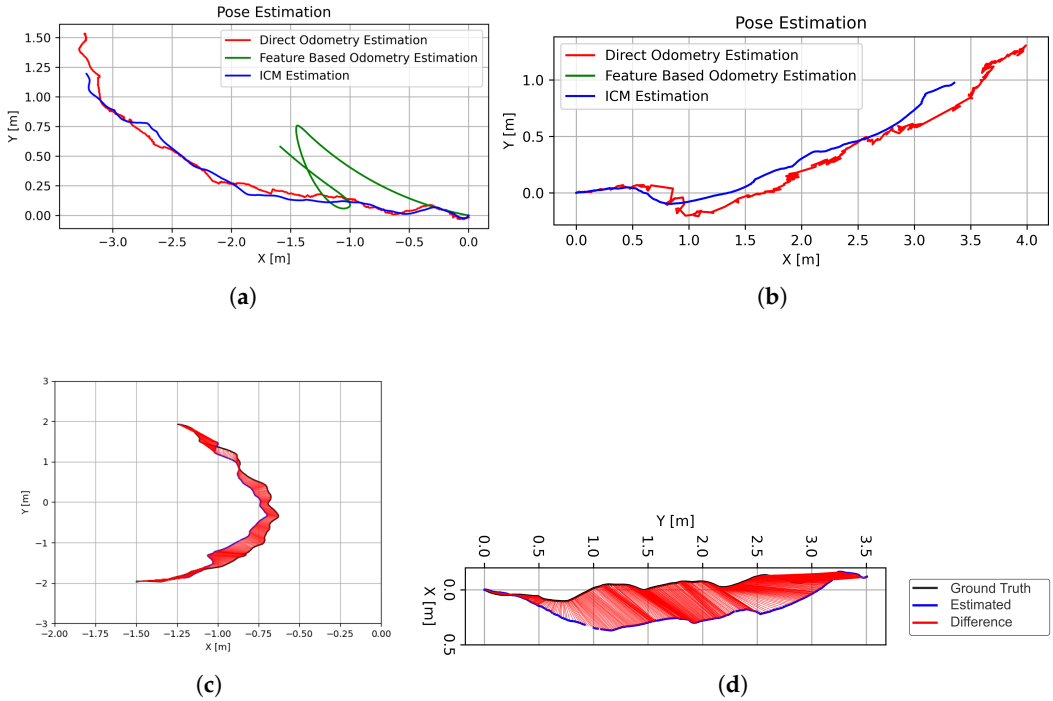



**Figure 6.** Internal trajectories recovered by the VO systems and the ICM module for the TUM RGB-D sequences: (**a**) *fr3/structure–no texture*. (**b**) *fr3/no structure–no texture*. (**c**,**d**) Aligned ground truth and estimated trajectories for the *fr3/structure–no texture* and *fr3/no structure–no texture* sequences, respectively. Note that, even when feature-based odometry fails, the system is still able to perform pose estimation due to the adaptable gain scheme.

The *fr3/no structure–no texture* sequence represents a more extreme case. Almost no visual information is provided during the trajectory of the camera. The feature-based module consistently fails to obtain a pose estimate due to the complete absence of features (Figure 6b). In contrast, the direct method remains functional. Due to the contribution of the direct VO module and the proposed fusion system, tracking failure is successfully avoided, yielding a 0.3361 m ATE RMS error for the estimated trajectory. Figure 6d shows a comparison between the ground truth and estimated trajectories after alignment.

To date, the estimation error was assessed using a standard metric that measures the deviation between the computed trajectory and the ground truth. However, this metric does not allow for a qualitative comparison between the initial trajectory (iteration 0) and the final optimized trajectory (after N iterations). To address this, we need a metric sensitive to the properties optimized by the potential functions, which can effectively evaluate the improvements made over the N iterations. Thus, we propose using the point-wise mean cost computed at initialization and at the final optimization iteration:

$$\bar{C} = \frac{1}{N_c} \sum_{i=1}^{N_c} C_i \tag{20}$$

The equation defines $\bar{C}$ as the average of the individual costs $C_i$ for all $N_c$ poses. Table 2 presents the calculated mean cost for the TUM RGB-D sequences at iteration 0 and after N iterations.

**Table 2.** Point-wise mean cost for de TUM RGB-D sequences.

| Sequence | Mean Cost Value |
|---|---|
| *fr1/plant it 0* | $3.8419 \times 10^{-4}$ |
| *fr1/plant it N* | $7.0364 \times 10^{-5}$ |
| *fr2/desk it 0* | $5.0571 \times 10^{-5}$ |
| *fr2/desk it N* | $8.2947 \times 10^{-6}$ |
| *fr3/s-t it 0* | $1.5861 \times 10^{-4}$ |
| *fr3/s-t it N* | $3.8727 \times 10^{-6}$ |
| *fr3/s-Nt it 0* | $1.0226 \times 10^{-4}$ |
| *fr3/s-Nt it N* | $4.6506 \times 10^{-6}$ |
| *fr3/Ns-t it 0* | $7.5609 \times 10^{-4}$ |
| *fr3/Ns-t it N* | $8.4838 \times 10^{-6}$ |
| *fr3/Ns-Nt it 0* | $2.3016 \times 10^{-3}$ |
| *fr3/Ns-Nt it N* | $3.8195 \times 10^{-5}$ |

A review of the table reveals a consistent reduction in the point-wise mean cost across all sequences. For each listed sequence, the mean cost at iteration 'N' is observed to be lower than the cost at iteration 0. This reduction indicates that the objective functions' costs were effectively minimized by the optimization process. The consistent decrease demonstrates that providing the system with information about past and future time steps leads to an improvement in trajectory estimation.

### *3.2. Field Tests*

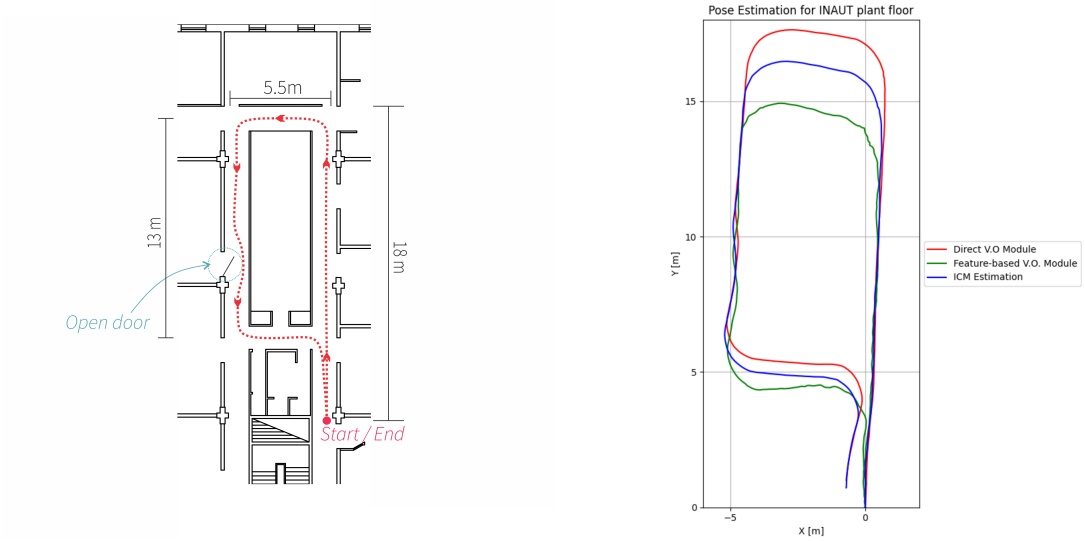
Real-time experiments were also conducted using an RGB-D Intel Realsense D435-i camera as the system input (more info in https://www.intel.la/content/www/xl/es/products/sku/190004/intel-realsense-depth-camera-d435i/specifications.html, accessed on 29 August 2025). This device features a more advanced depth estimation module compared to the one used in the TUM dataset, resulting in fewer missing depth values. Furthermore, it provides pre-aligned color images and depth maps, eliminating the need for additional preprocessing by the programmer.

#### 3.2.1. Indoor Test

For this experiment, the camera was mounted on a wheeled platform and driven through the INAUT facilities. The path consisted of a single closed loop, starting and ending at the same position, but with opposite orientations. Figure 7a shows a simplified floor plan of the INAUT with an approximate sketch of the executed trajectory, while Figure 7b allows for a comparison between the estimated trajectories from direct odometry, feature-based odometry, and ICM.

The output of the feature-based odometry module, shown in Figure 7b, exhibits noise throughout the trajectory. This noise is effectively suppressed by the ICM, likely due to the effect of adaptive odometry gains. The resulting trajectory (in blue) is also metrically consistent with the expected path and the floor layout.

On the left side of Figure 8, a top view of the reconstructed scene for this experiment is shown. The central part of the figure shows the incremental construction of the global map, and a zoomed-in section depicts the local map representation used for computations at that specific point in the trajectory.




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

**Figure 7.** (**a**) Approximated trajectory followed within the INAUT facilities. (**b**) Trajectory recovered by both VO systems and the ICM module (system output).

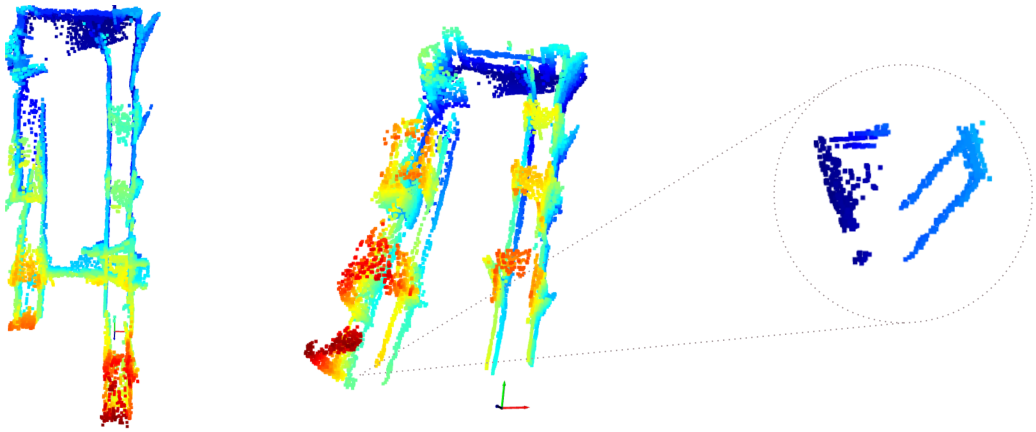


**Figure 8.** (**Left**): Final *global* map of the facilities of the Instituto de Automática after iteration 0 is completed. (**Center**): Snapshot of the *global* map at an intermediate stage of the initialization process. (**Right**): Detail of the *local* map **m**, built with previous observations.

In the structured office environment, both systems maintain pose estimation without tracking failure. Compared to the outdoor scenario, the consistent artificial lighting minimizes the brightness changes between frames, reducing potential interference in the estimation process.

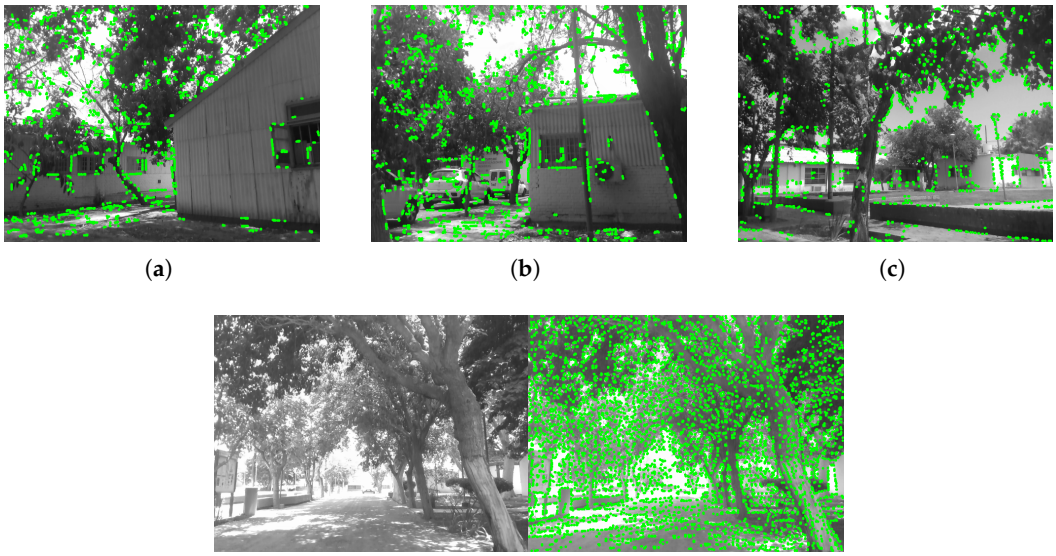
### 3.2.2. Outdoor Tests

The system's performance was also evaluated in outdoor environments. Navigation in such scenarios poses additional challenges such as varying lighting conditions (shadows and high brightness), uneven terrain, and dynamic elements. In this experiment, the camera was mounted on the front of a Sero Electric Sedán [35] electric vehicle, which traversed an internal road on the FI-UNSJ campus at an average speed of 2.7 m/s. The purpose of this experiment is to assess the system's ability to remain localized under

typical outdoor conditions, such as those previously discussed. Estimation accuracy was previously validated using benchmark datasets with available ground truth.

Among the two odometry systems that feed the ICM module, the direct method is the one most affected in these environments. This is primarily due to its reliance on the photometric consistency assumption, which is often violated in the presence of lighting variations. To mitigate these effects, the selected points for motion estimation must be carefully selected. A pixel selection algorithm described in Appendix A is responsible for this task. It operates under two key principles: giving priority (when available) to high-gradient regions that correspond to structural elements in the scene, and ensuring the spatial distribution of points across the image to best capture the camera's motion.

Sample frames from the outdoor experiments are shown in Figure 9. These grayscale images include green markers indicating the pixels selected for pose estimation. Each frame illustrates scenes with strong lighting contrast caused by sunlight passing through foliage and reflecting off building walls. If the algorithm were to select points solely based on gradient magnitude, they would be concentrated in areas of highest contrast—typically at the boundaries between bright sky and dark foliage. However, the system reduces the number of extracted points and prioritizes structured elements such as window frames, tree trunks, and lampposts, as can be seen in Figure 9a–c.



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

(**d**) (**e**)

**Figure 9.** Sample images from the outdoor tests. Figures (**a**–**c**) show how the internal pixel selection algorithm seeds points for pose estimation (green) under normal conditions. Figures (**d**,**e**) depict a very challenging scenario with very low gray-level difference in possible areas of interest and high gradient areas due to illumination changes.

This response is due to the internal use of texture-based criteria within the point extraction module, enabling it to disregard areas with rapid grayscale changes—such as foliage—that are less reliable for tracking. Nevertheless, due to the trade-off between performance and computational efficiency, the system may still select points in suboptimal regions during outdoor scenarios. Despite this, it consistently ensures a spatially balanced distribution of points and avoids selecting from low-texture areas such as sky, grass, or planar surfaces.

Real-world experiments may present challenging scenarios like that shown in Figure 9d. In addition to regions with high brightness, the areas that could potentially provide structural information exhibit very low gradients (i.e., nearly uniform grayscale

levels). In such scenarios, the system compensates by distributing selected points across the image, thereby preserving the tracking performance throughout the sequence.

The outdoor test itself involved following a round-trip path along one of the internal streets of the UNSJ campus, as illustrated in Figure 10a. The recovered trajectory after the test is shown in Figure 10b. Both odometry systems exhibit different biases when estimating the turning angle, which affects the final trajectory. However, since the proposed method combines these measurements with other sources of information, the proposed method yields a trajectory that is both metrically accurate and morphologically aligned with the actual path followed.

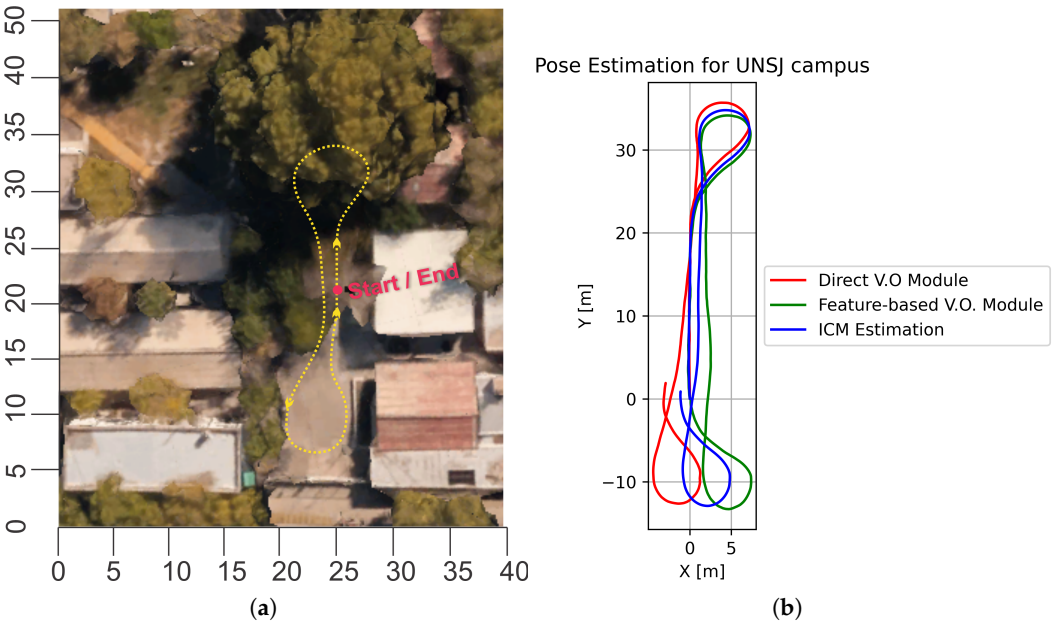



**Figure 10.** Outdoor tests: (**a**) Approximated trajectory followed by the electric vehicle on the UNSJ campus. (**b**) Results for ICM iteration 0.

## 4. Discussion

Most visual odometry systems reported in the state of the art have been designed to operate entirely within a single paradigm, either by means of direct methods or by means of indirect (feature-based) methods. The approach presented in this work was conceived to break such a dichotomy by employing both methods simultaneously within a unified framework, thereby taking advantage of the complementary benefits of each system while at the same time reducing their respective weaknesses. Special emphasis was placed on this latter point, in order to ensure the continuity of pose tracking even in situations in which one of the odometry systems failed completely.

In order to achieve this objective, a modular scheme with adaptable gains was implemented, whose values are automatically adjusted according to the estimation uncertainty provided by each of the odometry systems. The action of these gains makes it possible to regulate the influence of the potential functions associated with them during the optimization process performed by the ICM algorithm.

As illustrated by the results in Figure 6, the system demonstrates robustness against degradation or loss of the feature-based odometry. In such scenarios, the system adapts to feature-deprived environments by relying on the odometry obtained from the direct system. Even under these conditions, the recovered trajectory remains close to the ground truth.

Another challenge addressed during the evaluation of the system was the execution of field tests, whose principal aim was to validate the functioning of the system under a variety of lighting conditions. In the indoor office environment, artificial lighting remained

constant, while in the outdoor environment, certain areas presented high-contrast image regions. This latter condition is especially problematic for direct systems, because information can be drawn from high-gradient zones (for instance, where sunlight filters through foliage) that fail to reflect the actual structure of the image and thus impede accurate estimation. For that reason, an adaptive pixel selection scheme was implemented, which allows the system to search for those points in the image that best represent its structure. The results obtained confirm that incorporating such a scheme into the system offers a decisive advantage in preventing the direct system from losing localization. Overall, the results confirm that the proposed system is capable of maintaining localization in low-texture environments, which represent particularly adverse conditions for feature-based odometry systems.

The main limitation of the presented approach lies in the fact that ICM operates iteratively, commencing with an initial estimate of the environment map and robot trajectory. Subsequently, it refines these estimates through successive iterations until convergence to a solution is achieved. While these results represent an improvement over the initial estimates and can thus enhance the output of the system, ICM's performance is highly dependent on the quality of its initialization.

The main limitation of the proposed approach arises from the fact that the ICM algorithm operates in an iterative manner, beginning with an initial estimate of both the environment map and the robot trajectory. These estimates are subsequently refined through successive iterations until a convergence criterion is met. Although the iterative process consistently yields results that improve the initial estimates and thereby enhance the overall output of the system, the quality of the final solution remains strongly dependent on the accuracy of the initialization. Inadequate initial estimates could lead the algorithm to converge to suboptimal local maxima.

Future research directions include the incorporation of a loop closure detection system and the development of a potential function that allows the information obtained from such a system to be incorporated into the optimization process. Furthermore, although the present work focuses on the fusion of different visual odometry systems, it is of particular interest to carry out fusion tests with odometries coming from sensors of another nature, such as LiDAR sensors or wheel odometry from the robot.

## 5. Conclusions

This article introduces a novel modular fusion architecture designed for 6-DoF camera pose estimation, integrating outputs from both direct and feature-based visual odometry (VO) methods. Incoming RGB-D frames are concurrently processed by a feature-based VO module and a direct VO module, each generating independent pose estimations. These estimations are subsequently fed into the Iterated Conditional Modes (ICMs) algorithm as odometry potentials, which are then fused with a local map and a set of observations to yield a refined pose estimate.

The proposed novel design improves the system robustness against tracking loss in visually degraded or low-texture environments due to the effect of two key tools: first, a pixel selection strategy that ensures that the selected pixels for the direct VO module are those that best capture the camera's motion. Second, the system dynamically adjusts the influence of each odometry source based on its estimated uncertainty.

The comprehensive validation of the proposed method was conducted through experiments on the TUM RGB-D benchmark and real-world field tests in both indoor and outdoor settings, demonstrating its ability to maintain stable tracking in structured, semi-structured, and unstructured environments.

Despite its inherent computational efficiency and straightforward implementation, the ICM algorithm has not been widely utilized in visual mobile robot navigation ap-

plications. The contributions presented herein offer a foundation for deploying reliable autonomous localization and mapping solutions in practical mobile robotics, such as electric vehicles, where the tracking loss is unacceptable and environmental conditions are frequently challenging for conventional VO systems.

**Author Contributions:** Conceptualization, J.G. (Jeremías Gaia) and J.G. (Javier Gimenez); methodology, J.G. (Jeremías Gaia) and J.G. (Javier Gimenez); software, J.G. (Jeremías Gaia); validation, J.G. (Jeremías Gaia); formal analysis, J.G. (Javier Gimenez) and E.O.; investigation, J.G. (Jeremías Gaia); resources, F.R. and C.S.; data curation, J.G. (Jeremías Gaia); writing—original draft preparation, J.G. (Jeremías Gaia); writing—review and editing, E.O., F.R., F.U.-V. and C.S.; visualization, J.G. (Jeremías Gaia); supervision, C.S. and F.U.-V. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author.

**Acknowledgments:** Thank you to everyone who has helped with this research.

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

## Appendix A. Pixel Selection Algorithm

Developing an effective pixel selection strategy is vital for direct VO systems, as employing all image pixels for pose estimation in direct VO methods is computationally expensive. To this end, we introduce an adaptive selection method based on image texture indicators.

The main objective is to detect high-gradient regions that correspond to the structural elements of the scene, while avoiding image areas that do not provide relevant information for pose estimation. To achieve this, preprocessing techniques are applied to the input image, followed by the computation of local texture descriptors to assess the quality of different image regions.

### *Appendix A.1. Texture Descriptors*

The first image attribute targeted for detection is the absence of texture. Areas that exhibit uniform or nearly uniform gray-level values typically reflect low textural variation. The *Homogeneity* metric, also referred to as the inverse difference moment, is a statistical descriptor computed from the gray level co-occurrence matrix (GLCM) [36]. It is defined as

$$H = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{1}{1+(i-j)^2} p(i,j) \tag{A1}$$

where $p(i,j)$ denotes the $(i,j)$th element of the normalized gray-level co-occurrence matrix $P$, and $N_g$ is the number of gray levels in the image.

This metric quantifies the uniformity of gray levels, with higher values indicating larger areas of similar intensity and, therefore, a lower texture.

Another aspect needed to extract reliable points is to evaluate whether a given high-gradient area corresponds to a structural element or not. The gray level run length matrix (GLRLM) [37] quantifies the extent of consecutive runs of pixels that share the same gray level along a specific direction in the image. Two metrics can be derived from this matrix, the short run emphasis (SRE) and long run emphasis (LRE).

Given a GLRLM $V$, the element $v(i,j)$ represents the number of occurrences of a run of length $j$ at gray level $i$. Let $N_g$ be the number of gray levels, and $N_r$ be the number of run lengths considered. The total number of runs in the image, $N_{runs}$, is defined as

$$N_{runs} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} v(i,j) \tag{A2}$$

Based on this, the short run emphasis (SRE) and long run emphasis (LRE) are computed as follows:

$$SRE = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_r} \frac{v(i,j)}{j^2}}{N_{runs}} \tag{A3}$$

$$LRE = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_r} v(i,j) \cdot j^2}{N_{runs}} \tag{A4}$$

The SRE metric defined in Equation (A3) allows one to capture fine-grained variations in image texture. Specifically,

- High SRE values correspond to a predominance of short gray-level runs, which are indicative of the fine textures with frequent intensity transitions.
- Low SRE values reflect a higher occurrence of longer runs, suggesting a coarser texture with less frequent intensity changes.

In contrast, the LRE metric is designed to measure the presence of extended uniform regions within an image, where

- High LRE values signify a predominance of long gray-level runs, typical of coarse textures with large homogeneous areas or gradual transitions in intensity.
- Low LRE values indicate a larger proportion of short runs, which are characteristic of finer textures.

The combined analysis of this metrics offers a robust and comprehensive characterization of the textural properties of an image.

*Appendix A.2. Maximum Gradients Image*

Edges in an image correspond to locations where abrupt intensity transitions occur, typically indicating structural boundaries of objects within the scene. These edges are commonly detected by computing spatial derivatives, which highlight changes in gray levels. However, since the differentiation inherently amplifies high-frequency components, it can also enhance image noise, leading to spurious or fragmented edges. Therefore, applying noise suppression techniques prior to edge detection is essential to preserve the integrity of meaningful features.

In the context of pose estimation, edges play a critical role as they denote regions of high structural significance. Accurately identifying such features enables the robust tracking of camera motion. To this end, we employ a filtering approach known as the maximum gradients image, designed to emphasize prominent edges while suppressing weak and noisy responses. The process begins with the computation of the gradient magnitude $G$ from the input image. This is defined as

$$G = \sqrt{G_x^2 + G_y^2}, \tag{A5}$$

where $G_x$ and $G_y$ are the gradients in the horizontal and vertical directions, respectively. The resulting magnitude image is normalized to the range $[0, 255]$, and the threshold $Th_{curr}$ is

applied to retain only regions exhibiting sufficiently high gradient intensity. This step effectively removes low-contrast areas that do not contribute meaningful structural information.

To further refine the edge representation, we apply non-maximum suppression in two stages: first along the vertical direction and then along the horizontal. This technique ensures that only local maxima in the gradient field are preserved, enhancing the sharpness and spatial accuracy of the detected edges. Subsequently, morphological erosion is used to eliminate residual noise and to thin the edge contours, producing a binary image that highlights only the most salient edge points.

This entire process is independently applied to each channel of the input RGB image. Once the maximum gradient response is computed for each channel, the results are combined to generate the final maximum gradient image $G_{max}$. This fusion ensures that the edges prominent in any of the color channels are captured in the final output.

*Appendix A.3. Adaptive Reference Point Selection Algorithm*

The algorithm takes as input the RGB image and its grayscale counterpart. The RGB version is used to compute $G_{\max}$, while the grayscale image undergoes Gaussian blur filtering and Canny edge detection to produce a binary edge image.

Each image is then divided into patches of size $S_{\text{patch}}$ (initially set to the $S_{\text{patch}} = 16$), which are subsequently evaluated. At the end of the process, the number of extracted reference points is evaluated. If this number, denoted by $N_{\text{points}}$, does not meet a predefined minimum, the algorithm automatically adjusts either $S_{\text{patch}}$ or $Th_{\text{curr}}$ accordingly.

Appendix A.3.1. Adaptive Threshold Adjustment ($Th_{\text{curr}}$)

The threshold variable $Th_{\text{curr}}$ directly influences the number of high-gradient points in $G_{\max}$ that are eligible for selection as reference points. High values of $Th_{\text{curr}}$ restrict the appearance of gradients in $G_{\max}$, while low values allow low-gradient regions to emerge. An excessively high threshold can lead to significant information loss, reducing the number of potential keypoints. Conversely, an overly low threshold may result in the selection of low-gradient points, which negatively impact pose estimation. These trade-offs highlight the need for an adaptive thresholding mechanism.

At system initialization, $Th_{\text{curr}}$ is set to a default value. An initial keypoint extraction is performed using a fixed window size $S_{\text{patch}}$, and the resulting number of points is evaluated. If $N_{\text{points}}$ is insufficient, the threshold is decreased. The reduction percentage $r_{\text{perc}}$ is computed based on the homogeneity measures defined in Equation (A1), as follows:

$$r_{\text{perc}} = \frac{1}{\text{rows} \cdot \text{cols}} \sum_{i=1}^{\text{rows}} \sum_{j=1}^{\text{cols}} g(i,j) + \left( \frac{H^{0^\circ} + H^{90^\circ}}{2} \right) \tag{A6}$$

where $g(i,j)$ denotes the gradient magnitude at pixel $(i,j)$ in $G_{\max}$. High homogeneity values indicate that the threshold applied to generate $G_{\max}$ has been too restrictive and must be relaxed. Thus, $r_{\text{perc}}$ increases proportionally with the average homogeneity in the horizontal ($H^{0^\circ}$) and vertical ($H^{90^\circ}$) directions. The threshold is then updated as

$$Th_{\text{new}} = Th_{\text{curr}} \cdot r_{\text{perc}} \tag{A7}$$

A lower bound is defined for $Th_{\text{curr}}$, below which the threshold is no longer reduced; instead, the search window size $S_{\text{patch}}$ is modified.

Appendix A.3.2. Adaptive Search Window Adjustment ($S_{\text{patch}}$)

The value of $S_{patch}$ determines the size of the search window for reference points. In other words, it defines the dimensions of the patches extracted from each image. A fixed window size may lead to several issues:

- An excessively large window may result in too few extracted keypoints, failing to meet the minimum requirement.
- A window that is too small increases computational cost and may cause significant delays.
- In regions rich in texture or structure, a large window may underutilize the information, yielding few keypoints.
- Conversely, very small windows may lead to excessive keypoint extraction, slowing down subsequent pose estimation steps.

To address these issues, the window size is adaptively adjusted based on image characteristics. The algorithm processes the $G_{\text{max}}$, Canny edges, and grayscale versions of the input image, dividing them into patches of size $S_{\text{patch}}$. Each patch is further subdivided into four sub-patches of size $S_{\text{patch}}/2$ to ensure the spatial distribution of keypoints. The system is configured to extract a maximum of two keypoints per patch.

Each patch is individually evaluated to determine whether it meets the criteria for reliable keypoint selection. First, the grayscale patch is assessed using the SRE metric defined in Equation (A3). If $SRE_{\text{patch}} \geq 0.9$, the patch is discarded, as this indicates the presence of excessive gray-level variations, often corresponding to small or distant textures, which are unreliable for pose estimation.

Next, the binary patch derived from the Canny edge detector is analyzed using the LRE metric from Equation (A4). The Canny image is used due to its binarized nature, which facilitates the efficient computation of long edge sequences. If $LRE_{\text{patch}} \leq 200$, the patch is also discarded. Only patches satisfying both $SRE_{\text{patch}} < 0.9$ and $LRE_{\text{patch}} > 200$ are deemed suitable for keypoint extraction. These conditions ensure the presence of nearby objects or structured elements that contribute to accurate pose estimation.

If the desired number of keypoints is not obtained and $Th_{\text{curr}}$ has already reached its lower bound, $S_{\text{patch}}$ is halved. If this is still insufficient, the system attempts to extract three keypoints per patch to increase the likelihood of reaching the target. In extreme cases, illustrated in Figure 9e, where neither reducing the window size nor lowering $Th_{\text{curr}}$ is enough, the system extracts keypoints in a spatially uniform manner across the entire image, ensuring sufficient spatial coverage to capture camera motion.

The reference thresholds of 0.9 for SRE and 200 for LRE were empirically determined through experimentation, representing a trade-off between performance and computational efficiency that yields robust results in both indoor and outdoor environments.

## References


1. Yang, K.; Cheng, Y.; Chen, Z.; Wang, J. Slam meets nerf: A survey of implicit slam methods. *World Electr. Veh. J.* **2024**, *15*, 85. [CrossRef]
2. Lluvia, I.; Lazkano, E.; Ansuategi, A. Active Mapping and Robot Exploration: A Survey. *Sensors* **2021**, *21*, 2445. [CrossRef]
3. Placed, J.A.; Strader, J.; Carrillo, H.; Atanasov, N.; Indelman, V.; Carlone, L.; Castellanos, J.A. A survey on active simultaneous localization and mapping: State of the art and new frontiers. *IEEE Trans. Robot.* **2023**, *39*, 1686–1705. [CrossRef]
4. Gaia, J.; Orosco, E.; Rossomando, F.; Soria, C. Mapping the Landscape of SLAM Research: A Review. *IEEE Lat. Am. Trans.* **2023**, *21*, 1313–1336. [CrossRef]
5. Li, Y.; An, J.; He, N.; Li, Y.; Han, Z.; Chen, Z.; Qu, Y. A Review of Simultaneous Localization and Mapping Algorithms Based on Lidar. *World Electr. Veh. J.* **2025**, *16*, 56. [CrossRef]
6. Wu, D.; Ma, Z.; Xu, W.; He, H.; Li, Z. Visual Odometry Based on Improved Oriented Features from Accelerated Segment Test and Rotated Binary Robust Independent Elementary Features. *World Electr. Veh. J.* **2024**, *15*, 123. [CrossRef]
7. Scaramuzza, D.; Fraundorfer, F. Visual odometry [tutorial]. *IEEE Robot. Autom. Mag.* **2011**, *18*, 80–92. [CrossRef]

8. Tardif, J.P.; Pavlidis, Y.; Daniilidis, K. Monocular visual odometry in urban environments using an omnidirectional camera. In Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 22–26 September 2008; pp. 2531–2538. [CrossRef]
9. Bescos, B.; Fácil, J.M.; Civera, J.; Neira, J. DynaSLAM: Tracking, mapping, and inpainting in dynamic scenes. *IEEE Robot. Autom. Lett.* **2018**, *3*, 4076–4083. [CrossRef]
10. 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; pp. 834–849. [CrossRef]
11. Engel, J.; Stückler, J.; Cremers, D. Large-scale direct SLAM with stereo cameras. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 1935–1942. [CrossRef]
12. Engel, J.; Koltun, V.; Cremers, D. Direct sparse odometry. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 611–625. [CrossRef] [PubMed]
13. Zhu, Z.; Peng, S.; Larsson, V.; Xu, W.; Bao, H.; Cui, Z.; Oswald, M.R.; Pollefeys, M. Nice-slam: Neural implicit scalable encoding for slam. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 12786–12796.
14. 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. [CrossRef]
15. Yang, N.; Stumberg, L.v.; Wang, R.; Cremers, D. D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1281–1292. [CrossRef]
16. Matsuki, H.; Murai, R.; Kelly, P.H.; Davison, A.J. Gaussian splatting slam. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 18039–18048.
17. Schlegel, D.; Grisetti, G. Visual localization and loop closing using decision trees and binary features. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Republic of Korea, 9–14 October 2016; pp. 4616–4623. [CrossRef]
18. Nobis, F.; Papanikolaou, O.; Betz, J.; Lienkamp, M. Persistent Map Saving for Visual Localization for Autonomous Vehicles: An ORB-SLAM 2 Extension. In Proceedings of the 2020 Fifteenth International Conference on Ecological Vehicles and Renewable Energies (EVER), Monte-Carlo, Monaco, 10–12 September 2020; pp. 1–9. [CrossRef]
19. Elvira, R.; Tardós, J.D.; Montiel, J.M. ORBSLAM-Atlas: A robust and accurate multi-map system. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 6253–6259. [CrossRef]
20. Li, J.; Wang, X.; Li, S. Spherical-model-based SLAM on full-view images for indoor environments. *Appl. Sci.* **2018**, *8*, 2268. [CrossRef]
21. Campos, C.; Elvira, R.; Rodríguez, J.J.G.; Montiel, J.M.; Tardós, J.D. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM. *IEEE Trans. Robot.* **2021**, *37*, 1874–1890. [CrossRef]
22. 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]
23. Folkesson, J.; Christensen, H. Graphical SLAM-a self-correcting map. In Proceedings of the IEEE International Conference on Robotics and Automation, New Orleans, LA, USA, 26 April–1 May 2004; Proceedings, ICRA'04; Volume 1, pp. 383–390.
24. Grisetti, G.; Kümmerle, R.; Stachniss, C.; Burgard, W. A tutorial on graph-based SLAM. *IEEE Intell. Transp. Syst. Mag.* **2010**, *2*, 31–43. [CrossRef]
25. Akai, N.; Akagi, Y.; Hirayama, T.; Morikawa, T.; Murase, H. Detection of localization failures using Markov random fields with fully connected latent variables for safe LiDAR-based automated driving. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 17130–17142. [CrossRef]
26. Kindermann, R.; Snell, J.L. *Markov Random Fields and Their Applications*; American Mathematical Society: Providence, RI, USA, 1980; Volume 1.
27. Jadaliha, M.; Choi, J. Fully Bayesian simultaneous localization and spatial prediction using Gaussian Markov random fields (GMRFs). In Proceedings of the 2013 American Control Conference, Washington, DC, USA, 17–19 June 2013; pp. 4592–4597.
28. Wang, C.; Li, T.; Meng, M.Q.H.; De Silva, C. Efficient mobile robot exploration with Gaussian Markov random fields in 3D environments. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 5015–5021.
29. Gimenez, J.; Amicarelli, A.; Toibero, J.M.; di Sciascio, F.; Carelli, R. Iterated conditional modes to solve simultaneous localization and mapping in Markov random fields context. *Int. J. Autom. Comput.* **2018**, *15*, 310–324. [CrossRef]

30. Gimenez, J.; Amicarelli, A.; Toibero, J.M.; di Sciascio, F.; Carelli, R. Continuous probabilistic SLAM solved via iterated conditional modes. *Int. J. Autom. Comput.* **2019**, *16*, 838–850. [CrossRef]
31. Besag, J. On the statistical analysis of dirty pictures. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **1986**, *48*, 259–279. [CrossRef]
32. Jaime, F.; Marzola, T.; Baldoni, M.; Mattoccia, S. HoloSLAM: A novel approach to virtual landmark-based SLAM for indoor environments. *SN Comput. Sci.* **2024**, *5*, 469.
33. Sturm, J.; Engelhard, N.; Endres, F.; Burgard, W.; Cremers, D. A benchmark for the evaluation of RGB-D SLAM systems. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 573–580.
34. Horn, B.K. Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Am.* **1987**, *4*, 629–642. [CrossRef]
35. Sero Electric Sedan Electric Vehicle. Available online: https://www.seroelectric.com/modelo-sedan/ (accessed on 7 May 2025).
36. Haralick, R.M.; Shanmugam, K.; Dinstein, I.H. Textural features for image classification. *IEEE Trans. Syst. Man, Cybern.* **1973**, *6*, 610–621. [CrossRef]
37. Galloway, M.M. Texture analysis using grey level run lengths. *Nasa Sti/Recon Tech. Rep. N* **1974**, *75*, 18555.

**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.