Article

Multi-Tag Fusion Localization Method Based on Geometric Constraints

Zhuojun Liu 1, Wengang Qin 2, Zexing Li 2 and Guofeng Zhou 1,*

1 College of Engineering, Shanghai Ocean University, Shanghai 201306, China; m210801339@st.shou.edu.cn
2 College of Mechanical and Power Engineering, Shanghai Jiao Tong University, Shanghai 200240, China; wengang.qin@sjtu.edu.cn (W.Q.); lizexing@sjtu.edu.cn (Z.L.)
* Correspondence: gfzhou@shou.edu.cn

Abstract: In environments where Global Navigation Satellite System (GNSS) signals are unavailable, our proposed multi-tag fusion localization method offers a robust solution for the precise positioning of vehicles or robots. During our research, we observed variations in the positioning information estimated from tags located at different positions within the same frame. Our goal was to extract reliable positioning information from this noisy data. By constructing geometric constraints, our method introduces an outlier factor to quantify the differences between tags. After effectively eliminating outliers, we enhanced the Kalman filter framework to accommodate the fusion of data from two or more tags, with the outlier factor dynamically adjusting the observation noise during the fusion process. The experimental results demonstrate that, even under the influence of motion and obstacles, our method maintains position errors within a 3 cm range and orientation errors within 3°. This indicates that our method possesses high positioning accuracy and stability.

Keywords: localization; visual tag; data fusion; geometric constraints; AprilTag

1. Introduction

Localization is fundamental to vehicle planning, control, and subsequent tasks, essential for the integrity of vehicle functions [1–3]. The most common positioning technology is Global Navigation Satellite System (GNSS); however, not all scenarios offer ample and reliable signals, such as urban canyons, parking lots, tunnels, and indoor areas [4], [5]. Visual tag positioning technology has become increasingly prevalent across a spectrum of applications, including robotics [6,7] and Unmanned Aerial Vehicles (UAVs) [8,9], offering innovative solutions for the precise positioning of intelligent transport systems. By employing tags with predefined coordinate information, this technology achieves centimeter-level accuracy in positioning, which is crucial for various mobile platforms, including but not limited to automotive applications.

Despite the low cost and capability for absolute positioning of visual tags [10,11], they still face issues of inaccuracy and instability. During the localization process utilizing tags, the detection quality is significantly influenced by the distance and rotation angle between the camera and the tag [12]. The precision of the system is also impacted by the number and position of the tags within the image. Furthermore, lighting conditions can affect the accuracy of tag detection, particularly for tags like AprilTag [13]. It is crucial that the tag size is substantial enough to offer adequate information while remaining within the camera’s field of view [14]. There are three main approaches to address these issues: improving the detection accuracy of individual tags, fusing individual tag data with other sensor information, and increasing the number of tag detections.

To enhance positioning accuracy, some researchers have focused on improving marker detection accuracy. Two processing techniques were proposed by [15], including a method that can filter out very inaccurate detections resulting from partial border
occlusion, and a new highly accurate method for edge refinement. Based on observations, ref. [16] proposed improvements include trigonometric yaw angle correction for camera alignment, real-time tag center tracking via a custom yaw-axis gimbal, and a novel pose-indexed probabilistic sensor error model for AprilTag utilizing Gaussian Processes and validated by particle filter tracking. In the process of vehicle movement, ref. [17] found that the marker detection error ranged from 0.028 m to 0.13 m. To reduce pose jitter, they designed a strategy to constrain the inner and outer corner points of the markers, limiting the relative error between markers to within 0.05 m.

Meanwhile, fusing marker information with other data can also improve positioning accuracy. Ref. [13] described an algorithm that improves the pose estimation accuracy of square fiducial markers in difficult scenes by fusing information from RGB and depth sensors. Ref. [18] utilized particle filtering techniques to fuse data from ArUco markers and sonar sensors, achieving effective integration of multi-source sensor information. Ref. [19] introduced a variational Bayesian estimator for mobile robot localization with unknown noise covariance, utilizing tags as landmarks and an onboard camera. By treating the unknown measurement noise covariance as a random matrix following an inverse Wishart distribution, the method simultaneously estimates the robot pose. Ref. [20] proposed an on-manifold extended Kalman filter for tag-based visual–inertial localization, addressing the topological structure of rotation and pose groups and associated uncertainty propagations. Ref. [21] combined the robustness of markers to environmental uncertainty with motion data to minimize the required number of tags. An Iterative Extended Kalman Filter (IEKF) is employed to fuse visual and motion information, and an area detection algorithm is introduced to accelerate AprilTag processing.

While single-tag localization techniques have seen significant advancements, they encounter challenges when marker observation is obstructed or incomplete during vehicle movement [22]. This highlights the necessity for a method that can integrate information from multiple markers, especially in environments with partial occlusions or dynamics. Ref. [23] designed an integrated vision and LIDAR robot localization method using two AprilTag tags for pose estimation, proving the stability of the proposed method over single-tag approaches.

The third category of methods, which we focus on, involves detecting multiple tags simultaneously and fusing their data. For example, Ref. [24] averages observations from multiple markers and applies a constant velocity motion model for Kalman Filtering (KF) but does not adequately account for variability among different markers. Ref. [17] integrates markers into SLAM and applies the 3σ principle to filter outliers, which may be insufficient with fewer than ten markers as it defaults to ranking pose estimates and selecting the median, potentially not reflecting true poses if the observation distribution is skewed. Ref. [25] fuses IMU data with multiple markers using Extended Kalman Filtering (EKF), which can negatively impact localization accuracy when kinematic predictions significantly deviate from actual conditions. The optimization algorithm presented in [26], fusing multiple ArUco marker data based on the Mahalanobis distance, shares conceptual similarities with our approach but also reports trajectory discontinuities due to marker detection errors, especially when markers are distant or partially occluded. In the aforementioned filtering methods, observation noise is often determined by trial and error. However, ref. [27] discovered that the observation noise of pose from planar fiducial markers varies across the measurement range, a fact that must be considered during sensor fusion for a reliable estimate. They present experimental measurements of fiducial markers in real and simulation scenarios for 2D pose estimation and propose analytical functions that approximate the variances of pose estimates.

This paper proposes a multi-tag fusion localization method based on geometric constraints. The approach begins with detecting visual markers to acquire relative pose information between the target and the markers. Following coordinate transformations, the global pose of the target in the world coordinate system is computed. To ensure data accuracy and reliability, geometric constraints are constructed for all observations,
effectively eliminating outliers. Subsequently, the filtered and reliable observations are incorporated into a multi-observation fusion framework utilizing a modified KF. This integration facilitates the effective fusion of multiple observations, ultimately leading to precise positioning information.

Our main contributions are as follows:

1. A linear fusion method using only a camera: This method is simple, low-cost, and avoids complexities and potential inaccuracies associated with motion model-dependent predictions.

2. Introducing an outlier factor based on geometric constraints: The outlier factor quantifies the variability among different tags, excluding outliers before fusion and dynamically adjusting observation noise during the fusion process, reducing the likelihood of initial noise settings being unreasonable.

The paper consists of several parts. Section 2 introduces common visual tags and provides a detailed description of the AprilTag’s detection process. Section 3 describes the outlier handling method before data fusion, focusing on the construction of geometric constraints. Section 4 outlines the fusion approach, utilizing a modified KF for integrating data from multiple tags. Section 5 summarizes the experimental setup and presents the results and analysis. The final section concludes this paper with a summary of the key findings.

2. Visual Tag Process

Visual tags serve as objects in the image that provide key reference points and measurement points, and they have a known size and shape. Common visual tags include AR-Tag [28], AprilTag [29], ArUco [30], and Stag [31], as shown in Figure 1. Ref. [32] conducted an experimental comparison of these four tags, evaluating them comprehensively in terms of accuracy, recognition rate, and computational cost. The experimental results showed that AR-Tag has a low computational cost but performs poorly in the detection rate of a single marker; AprilTag, Stag, and ArUco exhibit excellent performance in terms of direction and position; however, AprilTag has a high computational cost and is sensitive to motion blur; Stag and ArUco are sensitive to smaller marker sizes and larger distances; and the computational cost of ArUco increases with the number of markers. Given the diversity of visual tags and their differences in characteristics, it is necessary to weigh the selection of tags based on the needs and scenarios of the actual application.

2.1. AprilTag3 Detection

In the pursuit of a pose estimation method well-suited for our application, AprilTag3 [33] has been identified as the optimal choice, predominantly attributable to its outstanding capabilities in delivering precise orientation results and elevated detection rates. AprilTag3, a sophisticated iteration within the AprilTag family, stands out for its open-source accessibility, high real-time processing capabilities, and enhanced accuracy.

This advanced version has undergone significant optimization, resulting in a more robust algorithmic framework. Notably, these enhancements have led to substantial gains in computational efficiency and a reduction in memory usage, making AprilTag3 an
attractive proposition for resource-intensive applications. Additionally, AprilTag3 supports an innovative format of two-dimensional codes, expanding its utility and applicability.

As depicted in Figure 2, the algorithmic process begins with an image filtering step that downscales the initial image to accelerate subsequent operations. This is followed by adaptive thresholding, which generates a binarized image that clearly delineates the tag boundaries against the background.

The subsequent edge detection phase employs the union-find algorithm to segment the image into meaningful components, effectively filtering out irrelevant sections and thus enhancing the precision of the detection process. Quadrilateral fitting then identifies potential tags from clustered line segments through a calculated sequence of operations.

To address perspective distortion, a perspective correction step is integrated, utilizing a transformation matrix to reconcile the disparity between the tag’s anticipated and actual poses. The final stage involves image sharpening to mitigate blurriness, enabling the clear decoding of the tag ID and the determination of the tag’s relative position to the camera.

**2.2. Coordinate Transformation**

To ascertain the global coordinates of a target within a scene, it is imperative to perform transformations across multiple coordinate systems. This process engages several defined coordinate systems: the world coordinate system (W), tag coordinate system (T), camera coordinate system (C), and the self or robot coordinate system (M), all adhering to the right-hand rule for orientation. As illustrated in Figure 3, the X-axis is represented in red, the Y-axis in green, and the Z-axis in blue.
Firstly, the relative pose between the tag and the camera, denoted as $\mathbf{P}_C$, is transformed into the relative pose between the tag and the target, denoted as $\mathbf{P}_M$:

$$\mathbf{P}_M = \mathbf{R}_C^M \mathbf{P}_C + \mathbf{T}_C^M$$  \hspace{1cm} (1)$$

In the equation, $\mathbf{R}_C^M$ represents the rotation matrix from the self-coordinate system to the camera coordinate system, and $\mathbf{T}_C^M$ is the installation position of the camera. Subsequently, the coordinates of the target $\mathbf{P}$ in the tag coordinate system are obtained as $\mathbf{P}_T$:

$$\mathbf{P}_T = \mathbf{R}_C^T \mathbf{P}_M + \mathbf{T}_C^T$$  \hspace{1cm} (2)$$

$\mathbf{R}_C^T$ is the rotation matrix between the tag coordinate system and the camera coordinate system. During the initial deployment of the tags, each tag is assigned a unique identifier, and the world coordinates of the center of the tag are represented by the matrix $\mathbf{T}(id)$:

$$\mathbf{T}(id) = [x, y, z]^T$$  \hspace{1cm} (3)$$

Finally, the coordinates of the target $\mathbf{P}$ in the world coordinate system are calculated as $\mathbf{P}_W$:

$$\mathbf{P}_W = \mathbf{R}_W^T \mathbf{P}_T + \mathbf{T}(id)$$  \hspace{1cm} (4)$$

In this formula, $\mathbf{R}_W^T$ is the rotation matrix from the tag coordinate system to the world coordinate system. The aforementioned rotation matrices $\mathbf{R}_C^M$, $\mathbf{R}_C^T$, $\mathbf{R}_W^T$ are derived from the Equation (5):

$$\mathbf{R} = \mathbf{R}_x \mathbf{R}_y \mathbf{R}_z$$  \hspace{1cm} (5)$$

$\mathbf{R}_x$, $\mathbf{R}_y$, and $\mathbf{R}_z$, are the rotation matrices calculated from the corresponding angles, $\phi$, $\theta$, and $\psi$ representing the rotation around the X-axis, Y-axis, and Z-axis, respectively.

$$\mathbf{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix}$$  \hspace{1cm} (6)$$

$$\mathbf{R}_y = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix}$$  \hspace{1cm} (7)$$

$$\mathbf{R}_z = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$  \hspace{1cm} (8)$$

An image containing N tags, once processed through tag detection, yields N distinct relative poses. After undergoing coordinate transformation, these relative poses result in N global poses of the target in the world coordinate system at the same moment. These global poses are then utilized as observation values for subsequent fusion steps. The
observation values include the two-dimensional coordinates in the world coordinate system and the orientation angle.

3. Construction of Geometric Constraints

When multiple tags are detected in an image, the positioning data extracted from different tags possess varying degrees of uncertainty. This implies that some observed values may have significant errors, and these outliers can substantially affect the final positioning outcome. As depicted in Figure 4, at the same moment, observed values \(i, j, \) and \(k\) are obtained. Under ideal circumstances, observed values \(i, j, \) and \(k\) should coincide with the true position. However, due to factors such as location and lighting, deviations occur between the observed values and the actual position. In statistical terms, an outlier is defined as one or several values within a dataset that differ significantly from the rest, constituting abnormal data. This paper constructs geometric constraints to measure the variability among observed values, thereby identifying and eliminating outliers.

Figure 4. Schematic representation of the geometric constraints.

Consider any two observed values, \(i\) and \(j\). The coordinates of observed value \(i\) in the world coordinate system are \([x_i, y_i, z_i]\), and the angle it makes with the X-axis is \(\angle a\). The coordinates of observed value \(j\) in the world coordinate system are \([x_j, y_j, z_j]\), and the angle it makes with the X-axis is \(\angle b\).

In the XOY plane, the difference in distance between any two observed values, \(l_{ij}\), can be calculated as

\[
l_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}
\]  

(9)

To calculate the angular difference \(d_{ab}\), first determine the angles \(\angle a\) and \(\angle b\):

\[
\begin{align*}
a &= \arctan\left(\frac{y_i}{x_i}\right) \\
 b &= \arctan\left(\frac{y_j}{x_j}\right)
\end{align*}
\]

(10)

Then, find the difference in angles:

\[
d_{ab} = |a - b|
\]

(11)
If there are a total of N observed values, by combining them in pairs to construct geometric constraints and calculating the corresponding distances $l_{ijl}$ and angular differences $d_{ijd}$, $G(N)$ combinations are needed in total. The number of combinations is given by the binomial coefficient:

$$G(N) = C_N^2 = \frac{N!}{(N-2)!2}$$

(12)

Each observed value will participate in $(N - 1)$ geometric constraint constructions, resulting in $(N - 1)$ different $l_{ijl}$ and $d_{ijd}$. The sum of the distance differences for each observed value with respect to the other observed values is

$$l_i = \sum_{j=1}^{N-1} l_{ij}$$

(13)

The sum of the angular differences for each observed value with respect to the other observed values is

$$d_i = \sum_{j=1}^{N-1} d_{ij}$$

(14)

An outlier factor, $S(i)$, is introduced and defined using a normalization method based on the maximum and minimum values to describe the variability between observed values from both distance and angular differences:

$$S(i) = k_1 \frac{l_i - \min(l_i)}{\max(l_i) - \min(l_i)} + k_2 \frac{d_i - \min(d_i)}{\max(d_i) - \min(d_i)}$$

(15)

In the equations, the weight coefficient $k_1$, $k_2$ is used, which must satisfy

$$k_1 + k_2 = 1$$

(16)

$$\max(l_i) = \max[l_1, \ldots, l_{N-1}]$$

$$\min(d_i) = \min[d_1, \ldots, d_{N-1}]$$

(17)

4. Fusion of Multiple Observations

In the context of multi-tag localization, the identification and removal of aberrant observations is a fundamental step. Yet, the integration of the subsequent reliable observations from multiple tags is an equally significant challenge. Traditional KF has long been a cornerstone for data fusion, operating under the assumption that the motion of the target conforms to specific patterns, such as uniform velocity or constant acceleration. This assumption facilitates the formulation of kinematic equations that predict system states.

Within this established framework, observational data from detected tags are merged with these predicted states to enhance the accuracy of the localization process. However, the standard KF is limited in its capacity to integrate only two elements: the predicted motion and the actual observation. This limitation becomes particularly evident when attempting to incorporate the wealth of data from multiple reliable observations, a scenario that surpasses the filter’s original design.

To address this, a novel approach has been introduced. This approach diverges from the standard method by performing predictive updates based solely on the available observational data. This enhanced method is capable of handling an increased number of observations through an iterative process, which, in turn, refines the precision and stability of the localization system. The flow of this algorithm is illustrated in Figure 5.
At time $t$, $N$ tags are observed, and after constructing geometric constraints, $n$ reliable observations are as follows: $Z_1, Z_2, ..., Z_n$. The system state at time $t$ is $X_t$, and the system state at the $k$-th fusion is $X_k$, where $k \in [1, n-1]$. A total of $n-1$ fusions are required for each moment. The system state includes the two-dimensional coordinates in the world coordinate system and the orientation angle:

$$X_t = [x, y, \psi]^T$$  \hspace{1cm} (18)

$$X_k = [x, y, \psi]^T$$  \hspace{1cm} (19)

The relationship between $X_t$ and $X_k$ is given by:

$$X_t = X_k + w_{k-1}$$  \hspace{1cm} (20)

In this equation, $w_k$ represents the process noise in the fusion, which is normally distributed, $w_k \sim (0, Q)$, where $Q$ is the covariance matrix of the process noise.

The relationship between the observation values and the system state is

$$Z_i = HX_k + v_i$$  \hspace{1cm} (21)

Here, $H$ is the observation matrix, and $v_i$ represents the measurement noise of the $i$-th reliable observation, which is also normally distributed, $v_i \sim (0, R)$, $i \in [1, n]$, where $R$ is the covariance matrix of the measurement noise.
At the k-th fusion, where $k \in [1, n-1]$, the fused value from the (k-1)-th fusion, $\hat{X}_{k-1}$ is used as the current moment’s predicted value, $\bar{X}_k$:

$$\bar{X}_k = \hat{X}_{k-1}$$  \hspace{1cm} (22)

The initial value, $\bar{X}_1$, is given by:

$$\bar{X}_1 = H^{-1}Z_1$$  \hspace{1cm} (23)

The prediction error, denoted as $\bar{e}_k$, is calculated as

$$\bar{e}_k = X_k - \bar{X}_k$$
$$= X_k - \hat{X}_{k-1}$$
$$= X_{k-1} + w_{k-1} - \hat{X}_{k-1}$$
$$= e_{k-1} + w_{k-1}$$  \hspace{1cm} (24)

The prediction error covariance matrix $\bar{P}_k$ is computed as

$$\bar{P}_k = E[\bar{e}_k\bar{e}_k^T]$$
$$= P_{k-1} + Q_{k-1}$$  \hspace{1cm} (25)

where $Q$ is the process noise covariance matrix.

The initial value, $\bar{P}_1$, is given by

$$\bar{P}_1 = P_{1-1}$$  \hspace{1cm} (26)

The observation noise is adjusted based on the outlier factor of the observation values:

$$R_i = (S(k) - S(1))R_i$$  \hspace{1cm} (27)

Here, $S(1)$ is the outlier factor of the first reliable observation, and $R_i$ is the initial observation noise. Since the reliable observations are sorted in ascending order based on the outlier factor, the first reliable observation has the smallest outlier factor value. This implies that a larger outlier factor corresponds to greater observation noise.

The Kalman gain $K_k$ is calculated as

$$K_k = \frac{\bar{P}_k H^T}{HP_k^T H + R_k}$$  \hspace{1cm} (28)

The new observation value $Z_{i+1}$ is introduced as the k-th observation

$$Z_k = Z_{i+1}$$  \hspace{1cm} (29)

The prediction is updated using the Kalman gain and the new observation value:

$$\hat{X}_k = \bar{X}_k + K_k (Z_k - H\bar{X}_k)$$  \hspace{1cm} (30)

Update the error:

$$e_k = X_k - \hat{X}_k$$
$$= X_k - (\bar{X}_k + K_k (Z_k - H\bar{X}_k))$$
$$= (I - K_k H)(X_k - \bar{X}_k) - K_k e_k$$  \hspace{1cm} (31)
where $I$ is the identity matrix.

Update the error covariance matrix:

$$P_x = E[e_x^T e_x] = (I - K_x H) \overline{P}_x$$

(32)

The iteration continues until all reliable observations have been integrated. At the end of the iteration, the final fused value at time $t$, $\hat{X}_t$, is obtained. At this point, the error covariance matrix will serve as the predicted error covariance matrix for the next moment $t + 1$:

$$\hat{X}_t = \hat{X}_{t-1}$$

(33)

$$\overline{P}_{t+1} = P_t = P_{t-1}$$

(34)

5. Experiment

In this section, we detail the experimental approach undertaken to validate the efficacy of our multi-tag fusion localization method. Utilizing the Isaac Sim platform, we have conducted simulation-based experiments that offer a cost-effective and flexible framework for assessing our method. The decision to employ simulation was driven by the constraints of time and budget, while also allowing for a diverse range of scenario configurations through parameter adjustments. This approach enables us to systematically test the robustness and accuracy of our method under controlled conditions. Our primary objectives were twofold: to initially validate the geometric constraints by demonstrating their effectiveness in filtering out outliers and subsequently to assess the overall positioning accuracy and stability of the proposed localization method. These objectives guide the design of our experiments and the analysis of the results obtained.

5.1. Experimental Setup

The Isaac Sim camera model, depicted in Figure 6, generates high-fidelity images for reliable testing. The reliability of the simulation camera was validated through fixed-point tests, confirming its ability to provide trustworthy and high-quality images. Positioned at the rear of the vehicle at a height of 1.15 m, the camera captured 6 to 8 tags per frame at a resolution of 1280 × 720 pixels and a frame rate of 25 fps.
This study designed multiple simulation scenarios, as depicted in Figure 7.

Simulation Scenario 1: Visual tags are laid out on the ground. AprilTag’s tag36 h 10 series is used for the visual tags, with each tag measuring 0.04 m × 0.04 m and spaced 0.15 m apart. This initial setup establishes a baseline for evaluating the method’s accuracy in an unobstructed environment.

Simulation Scenario 2: Adds obstacles based on Scenario 1. The incorporation of obstacles is intended to simulate real-world conditions where line-of-sight to tags may be partially blocked, thereby testing the method’s resilience to occlusions.

Simulation Scenario 3: Similar to Scenario 1, but with tags measuring 0.03 m × 0.03 m and spaced 0.1 m apart. Reducing the tag size and adjusting the spacing allows us to examine the method’s sensitivity to changes in tag density and visibility range.

Simulation Scenario 4: Adds obstacles based on Scenario 3. Further introducing obstacles with the smaller tags assesses the method’s adaptability to challenging conditions with increased occlusion and reduced tag size.
This study conducted a series of six experiments designed to evaluate the effectiveness of geometric constraints and to assess the positioning accuracy and stability of the proposed method, as detailed in Table 1. The simulation software provided the ground truth for the experiments, establishing a reliable benchmark for comparison.

Due to the uniform scaling of the tag models, changes in tag size were accompanied by corresponding adjustments in the spacing between tags. As depicted in Figure 7, the simulation software offered realistic lighting conditions, resulting in varying illumination for tags at different locations. This feature of the simulation software introduced an additional layer of realism to the experiments, as it mimicked the diverse lighting conditions that tags might encounter in real-world scenarios.

The inclusion of these factors in the experimental design allowed for a more comprehensive assessment of the proposed method’s performance under conditions that closely resemble real-world applications. This attention to detail ensures that the results obtained from the simulation are not only indicative of the method’s efficacy but also reflective of its robustness in varying environmental conditions.

Table 1. Experimental setup.

<table>
<thead>
<tr>
<th>Group</th>
<th>State of Motion</th>
<th>Tag Size</th>
<th>Tag Interval</th>
<th>Whether Obstructions Exist</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>still</td>
<td>0.04 m × 0.04 m</td>
<td>0.15 m</td>
<td>no</td>
</tr>
<tr>
<td>2</td>
<td>moving</td>
<td>0.04 m × 0.04 m</td>
<td>0.15 m</td>
<td>no</td>
</tr>
<tr>
<td>3</td>
<td>moving</td>
<td>0.04 m × 0.04 m</td>
<td>0.15 m</td>
<td>yes</td>
</tr>
<tr>
<td>4</td>
<td>still</td>
<td>0.03 m × 0.03 m</td>
<td>0.1 m</td>
<td>no</td>
</tr>
<tr>
<td>5</td>
<td>moving</td>
<td>0.03 m × 0.03 m</td>
<td>0.1 m</td>
<td>no</td>
</tr>
<tr>
<td>6</td>
<td>moving</td>
<td>0.03 m × 0.03 m</td>
<td>0.1 m</td>
<td>yes</td>
</tr>
</tbody>
</table>

5.2. Geometric Constraint Effectiveness

A critical aspect of this research was to demonstrate the effectiveness of geometric constraints in enhancing the accuracy of the positioning system.
5.2.1. Single-Frame Validation

For a demonstrative purpose, we extracted a specific frame from the simulation dataset, capturing both the true and observed positioning values. Within this frame, eight tags were discerned, with the identifiers {1120, 1121, 1136, 1137, 1408, 1409, 1410, 1425}. Implementing geometric constraints, we assigned a consistent weight coefficient of 0.5 to all observations, as per Equation (15), to ascertain the outlier factor for each. The results of these calculations are presented in Table 2.

After sorting the observations by their outlier factors in ascending order, we designated the top 4 observations with the lowest outlier factors as reliable. These reliable observations were identified by the IDs {1121, 1410, 1409, 1137}, while the remaining observations were considered outliers: {1425, 1408, 1120, 1136}.

Figure 8 presents a scatter plot of the observed tag coordinates within the world coordinate system, where each distinct point corresponds to a single observation identified by its unique tag ID. The positioning of these points in proximity to the true values illustrates the precision of our method, demonstrating the successful application of our filtering process in discerning reliable observations from outliers.

Table 2. Outlier factor.

<table>
<thead>
<tr>
<th>ID</th>
<th>Outlier Factor</th>
<th>Whether It Is a Reliable Observation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1121</td>
<td>0</td>
<td>yes</td>
</tr>
<tr>
<td>1410</td>
<td>0.04049</td>
<td>yes</td>
</tr>
<tr>
<td>1409</td>
<td>0.08078</td>
<td>yes</td>
</tr>
<tr>
<td>1137</td>
<td>0.25164</td>
<td>yes</td>
</tr>
<tr>
<td>1425</td>
<td>0.67949</td>
<td>no</td>
</tr>
<tr>
<td>1408</td>
<td>0.80974</td>
<td>no</td>
</tr>
<tr>
<td>1120</td>
<td>0.83909</td>
<td>no</td>
</tr>
<tr>
<td>1136</td>
<td>1</td>
<td>no</td>
</tr>
</tbody>
</table>

Figure 8. Schematic representation of the observed values.

Furthermore, Table 3 illustrates the variance of the observations before and after the removal of outliers. The variance in the X, Y, and yaw dimensions of the reliable observations is consistently lower than that of all observations combined. This reduction in variance demonstrates that the data become more concentrated and stable after the application of geometric constraints, confirming the reliability of the filtered observations.
Table 3. Comparison of variance before and after excluding outliers.

<table>
<thead>
<tr>
<th>Dimensionality</th>
<th>The Variance of All Observations</th>
<th>Variance of Reliable Observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>0.00715</td>
<td>0.00254</td>
</tr>
<tr>
<td>Y</td>
<td>0.00436</td>
<td>0.00184</td>
</tr>
<tr>
<td>Yaw</td>
<td>0.01124</td>
<td>0.00846</td>
</tr>
</tbody>
</table>

5.2.2. Process Validation

In this section, we compare the method without geometric constraints and our proposed method over a sustained period, with detailed results presented in Table 4. This comparison allows us to evaluate the impact of geometric constraints on the system’s accuracy and stability in a continuous positioning process.

The Method Without Geometric Constraints: This method fused all raw observations. It treated each tag’s data equally, applying a uniform and constant level of observation noise throughout the fusion process.

Our Method: The proposed method first culled outliers based on outlier factors, then dynamically adjusted observation noise for each tag’s data during fusion, enhancing accuracy and stability.

Table 4. Comparative error analysis.

<table>
<thead>
<tr>
<th>Group</th>
<th>Method Without Geometric Constraints</th>
<th>Our Method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Distance Error (m)</td>
<td>Yaw Error (rad)</td>
</tr>
<tr>
<td>1</td>
<td>0.02246</td>
<td>0.03816</td>
</tr>
<tr>
<td></td>
<td>0.00386</td>
<td>0.00089</td>
</tr>
<tr>
<td></td>
<td>0.01113</td>
<td>0.00989</td>
</tr>
<tr>
<td>2</td>
<td>0.16475</td>
<td>0.14654</td>
</tr>
<tr>
<td></td>
<td>0.00231</td>
<td>0.00014</td>
</tr>
<tr>
<td></td>
<td>0.05813</td>
<td>0.02442</td>
</tr>
<tr>
<td>3</td>
<td>0.61676</td>
<td>0.10386</td>
</tr>
<tr>
<td></td>
<td>0.01283</td>
<td>0.00321</td>
</tr>
<tr>
<td></td>
<td>0.05587</td>
<td>0.05908</td>
</tr>
<tr>
<td>4</td>
<td>0.05618</td>
<td>0.03842</td>
</tr>
<tr>
<td></td>
<td>0.00033</td>
<td>0.00025</td>
</tr>
<tr>
<td></td>
<td>0.01646</td>
<td>0.00837</td>
</tr>
<tr>
<td>5</td>
<td>0.09544</td>
<td>0.10569</td>
</tr>
<tr>
<td></td>
<td>0.00502</td>
<td>0.01421</td>
</tr>
<tr>
<td></td>
<td>0.04035</td>
<td>0.03189</td>
</tr>
<tr>
<td>6</td>
<td>0.14736</td>
<td>0.10569</td>
</tr>
<tr>
<td></td>
<td>0.00148</td>
<td>0.00113</td>
</tr>
<tr>
<td></td>
<td>0.07850</td>
<td>0.03613</td>
</tr>
</tbody>
</table>

The table provides a clear comparison of the error metrics between the method without geometric constraints and our proposed method, highlighting the impact of incorporating geometric constraints on positioning accuracy.

For positional error, the method without geometric constraints had an average error within 8 cm, but this error peaked at a maximum of 14 cm during the motion sequence. This variability suggests a potential lack of reliability in positioning accuracy.

In contrast, our method, which applies geometric constraints, showcased superior precision with a maximum average error of 2.611 cm and a maximum error capped at 2.968 cm. This represents a significant improvement in positional accuracy, demonstrating the advantage of using geometric constraints.

When it comes to orientation angle error, the method without geometric constraints exhibited a maximum error of 8.39° and a maximum average error of 3.38°. These figures indicate a higher likelihood of inaccuracy in orientation estimation.
However, our method achieved a maximum error of 2.90° and a maximum average error of just 1.99° in orientation angle estimation. This substantial enhancement in orientation accuracy further underscores the effectiveness of geometric constraints in refining the precision and stability of the positioning system.

We have selected a specific dataset to visually demonstrate the comparative performance between our proposed method and the method without geometric constraints. The visualization focuses on the X, Y, and YAW positioning data for both methods. As depicted in Figure 9, our method exhibits a smoother trajectory, indicative of more stable and precise localization. In contrast, the method lacking geometric constraints shows noticeable spikes and abrupt changes, revealing its susceptibility to inaccuracies.

![Figure 9. Method comparison with/without geometric constraints. (a) Trajectory; (b) X; (c) Y; (d) Yaw.](image)

5.3. Method Performance

In this section, we present graphical representations of our method’s performance, focusing on the trajectory and positional data from Groups 2 and 3 as outlined in the experimental setup table, where the positioning information obtained through our method is compared against the true positions provided by the simulation software.

Figure 10 (Group 2): This figure displays the raw observational data for the vehicle’s motion as a scatter plot, which clearly shows the substantial dispersion of points due to
the presence of outliers. Our method adeptly addresses these irregularities, producing a trajectory that closely mirrors the true path with a notable reduction in fluctuations.

Figure 10. Experimental results of Group 2. (a) Trajectory; (b) $x$; (c) $y$; (d) Yaw.

Figure 11 (Group 3): The motion trajectory of the vehicle is portrayed under more complex conditions, where movement is compounded by the presence of obstacles. The increased presence of outliers, particularly at corners where occlusions affect observation quality, is evident. Despite these challenges, our method demonstrates its robustness by producing a trajectory that is significantly smoother and more aligned with the true path compared to the raw data.
Based on the experimental setup outlined in Table 1 and the data presented in Table 4, we conclude the following:

Motion State: The motion state of the vehicle significantly affects positioning accuracy, with moving Groups 2 and 5 showing higher positioning errors compared to the stationary Groups 1 and 4. Our method, however, reduces these errors, bringing the positioning accuracy of moving states close to that of stationary states, indicating its effectiveness in handling motion challenges.

Tag Size: Comparisons across different tag sizes demonstrate that our method maintains low positioning errors regardless of tag size, whether larger (Groups 1 and 2) or smaller (Groups 4 and 5). This finding suggests that our method is compatible and consistent with various tag dimensions.

Obstacle Obstruction: The presence of obstacles in Groups 3 and 6, as opposed to the absence of obstructions in Groups 2 and 5, results in decreased positioning accuracy. However, our method’s identification and removal of outliers effectively lessen the impact of obstructions on positioning outcomes.

6. Conclusions
The present study offers a contribution to the field of multi-tag localization by proposing a method that incorporates geometric constraints to address the issues of instability and inaccuracy that may arise in such scenarios.

We have developed a method that calculates an outlier factor based on geometric constraints, which may provide a means to characterize the variability among observational data points from multiple tags. This approach could potentially enhance the accuracy of the positioning process by dynamically adjusting the observation noise for each tag after identifying and removing outliers. Our method also includes an extension to the traditional KF technique, designed to manage the fusion of data from multiple tags. This extension, which is based solely on observational data for prediction updates, may offer a pathway to bypass the limitations of conventional KF that typically integrate two data sources.

Preliminary results from high-fidelity simulations indicate that our method can achieve a reasonable level of positioning accuracy and stability, with errors maintained within a 3cm range for position and 3° for orientation. These results are encouraging and suggest that the method may be applicable in environments where GNSS signals are not available or are subject to interference. In our future work, we aim to integrate Z-axis constraints into our localization method to achieve six-degree-of-freedom (6-DOF) positioning. This enhancement will be crucial for high-precision navigation in environments with varying terrain. Additionally, we plan to deploy our method on a real vehicle, where we will tackle and overcome the practical issues encountered in actual deployment scenarios.

Author Contributions: Conceptualization, Z.L. (Zhuojun Liu) and Z.L. (Zexing Li); Data curation, Z.L. (Zhuojun Liu); Formal analysis, Z.L. (Zhuojun Liu); Funding acquisition, W.Q. and G.Z.; Investigation, Z.L. (Zhuojun Liu) and W.Q.; Methodology, Z.L. (Zhuojun Liu); Project administration, W.Q.; Resources, W.Q., Z.L. (Zexing Li), and G.Z.; Software, Z.L. (Zhuojun Liu) and Z.L. (Zexing Li); Supervision, W.Q. and G.Z.; Validation, Z.L. (Zhuojun Liu); Visualization, Z.L. (Zhuojun Liu); Writing—original draft, Z.L. (Zhuojun Liu) and G.Z.; Writing—review and editing, Z.L. (Zhuojun Liu), W.Q., and G.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

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

Acknowledgments: The authors would like to acknowledge the generous support of Qin Wengang, Li Zexing, and Zhou Guofeng in funding aspects of this work.

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

References


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.