Scientific Research Publishing

**Positioning, 2019, 10, 51-72**
https://www.scirp.org/journal/pos
ISSN Online: 2150-8526
ISSN Print: 2150-850X

# Simultaneous Localization and Mapping Solutions Using Monocular and Stereo Visual Sensors with Baseline Scaling System

**Akram Afifi*, Brendan Woo**

Faculty of Applied Sciences and Technology, Humber Institute of Technology and Advanced Learning, Toronto, Canada
Email: *akram.afifi@humber.ca, brendan.woo@humber.ca

**How to cite this paper:** Afifi, A. and Woo, B. (2019) Simultaneous Localization and Mapping Solutions Using Monocular and Stereo Visual Sensors with Baseline Scaling System. *Positioning*, **10**, 51-72.
https://doi.org/10.4236/pos.2019.104004

**Received:** September 10, 2019
**Accepted:** October 13, 2019
**Published:** October 16, 2019



## Abstract

In this paper, SLAM systems are introduced using monocular and stereo visual sensors. The SLAM solutions are implemented in both indoor and outdoor. The SLAM samples have been taken in different modes, such as a straight line that enables us to measure the drift, in addition to the loop sample that is used to test the loop closure and its corresponding trajectory deformation. In order to verify the trajectory scale, a baseline method has been used. In addition, a ground truth has been captured for both indoor and outdoor samples to measure the biases and drifts caused by the SLAM solution. Both monocular and stereo SLAM data have been captured with the same visual sensors which in the stereo situation had a baseline of 20.00 cm. It has been shown that, the stereo SLAM localization results are 75% higher precision than the monocular SLAM solution. In addition, the indoor results of the monocular SLAM are more precise than the outdoor. However, the outdoor results of the stereo SLAM are more precise than the indoor results by 30%, which is a result of the small stereo baseline cameras. In the vertical SLAM localization component, the stereo SLAM generally shows 60% higher precision than the monocular SLAM results.

## Keywords

SLAM, Vision, Kalman Filter, Monocular, Stereo

## 1. Introduction

Simultaneous localization and mapping (SLAM) is the procedure of building the map of the surrounding environment of a vehicle/rover and uses the computed map to determine the vehicle/rover location. In the past decade there was active

research to solve the SLAM problem using several methods of computations. The great majority of work has focused on improving computational efficiency while ensuring consistent and accurate estimates for the map and vehicle pose. However, there has also been much research on issues such as nonlinearity, data association, and landmark characterization, all of which are vital in achieving a practical and robust SLAM implementation. SLAM process challenges are centered on methods enabling large-scale implementations in increasingly unstructured environments and especially in the GPS denied environment. Indoor 3D mapping helps to view the three-dimensional objects and spatial structure on the computer efficiently. A precise description about the scene through the sensors is required. In the latest research contribution, SLAM has become more popular, there are three major categories: visual-based, laser-based and depth-visual based. There are the visual-based SLAM solutions such as ORB-SLAM [1] and LSD-SLAM [2], where only the image information has been used to develop the 3D map. As a result, several challenges had to be adjusted such as the scale-drift problem [3], and light change effect. The visual-based SLAM can be implemented using different sensors setup such as monocular, stereo, and multiple visual sensor. On the other hand, the visual-depth SLAM [4] [5] [6] utilizes an integrated depth sensor with visual sensor such as Microsoft Kinect, Intel real sense, and Xtion Pro. The visual-depth sensors usually have a limited scan angle and range which results in a SLAM computation challenge for the large environments [7] [8] [9] [10] [11]. Moreover, Laser-based SLAM can be computed in large number of environments with different scale due to the larger laser range. Commonly, SLAM can be computed using three paradigms namely: Kalman filters, Particle filters and Graph-based [10] [11] [12] [13].

Kalman filters have two main parts: prediction and update. In order to solve the nonlinear problem, the Extended Kalman Filter (EKF) was put forward. The EKF-SLAM can only deal with a single mode. It is successful in medium-scale scenes but when it comes to a large map, it becomes computationally intractable. Certainly, there is better method solving the nonlinear problem like Unscented Kalman Filter (UKF). The Kalman filter and its variants can only model Gaussian distributions, so an approach is needed to deal with the arbitrary distributions. However, particle filters can deal with the arbitrary distributions by using multiple samples. This method deems that the more particles fall into a region, the higher the probability of the region is [14]. The posterior probability is represented by a set of particles which have been weighted. The particle filters-based SLAM [7], models the vehicle/rover's path by sampling and computing the landmarks given the path. Graph-based SLAM [9] considers that a graph is composed of poses and constraints between poses. By constructing a graph to minimize the sum of the squared error, in fact, it is a method of optimization that uses linear methods to solve the non-linear problem. In this research, the EKF has been used following the approach of [1] [2] [3] [4] [5]; visual-based and visual-depth SLAM systems are presented. Based on the sensors setup and data processing the visual-based SLAM can be either single sensor,

(monocular) [10]-[17], or two visual sensors separated by a baseline (stereo), or more than two sensor with a setup with a scene overlap that eventually covers same target to complete the views and eliminate the blind spots [18] [19] [20] [21]. In this research, both monocular and stereo systems have been used to provide a visual-based SLAM solution. In addition, a visual-depth based SLAM is presented using stereo depth sensors along with a visual monocular sensor [7] [8] [9].

In this paper, SLAM system is introduced using monocular, and Stereo visual sensors. SLAM solutions are implemented in both indoor and outdoor using extended Kalman filter. The SLAM samples have been taken in different modes such as a straight line that enable us to measure the drift in addition to the loop sample that is used to test the closure and its corresponding trajectory deformation. In order to verify the trajectory scale, a baseline method has been used. In addition, a ground truth has been captured for all the samples indoor and outdoor to measure the biases and drifts caused by the SLAM solution. Both monocular and stereo SLAM data have been captured with the same visual sensors which in the stereo situation had a baseline of 20.00 cm. It has been shown that, the stereo SLAM localization results are 75% higher precision than the monocular SLAM solution. In addition, the indoor results of the monocular SLAM are more precise than the outdoor. However, the outdoor results of the stereo SLAM are more precise than the indoor results by 30%, which a result of the small stereo baseline cameras. In the vertical SLAM localization component, the stereo SLAM generally shows 60% higher precision than the monocular slam results.

## 2. Slam Related Work

The state-based formulation of the SLAM includes the computation of a joint state made up of a vehicle/rover pose and the locations of captured landmarks. This problem formulation has a unique structure; the process model only affects vehicle/rover pose states while the observation model only makes reference to a single vehicle-landmark pair. A large range of strategies have been developed to take advantage of this special structure in limiting the computational complexity of the SLAM algorithm. There are two categorize of the SLAM techniques that aims for improving the computational efficiency namely; optimal or conservative solutions. The optimal solutions target to reduce required computation and resulting in computations and covariances for the full-form SLAM algorithm. While the conservative algorithms result in estimates that have larger uncertainty or covariance than the optimal result. Usually, conservative algorithms are less accurate but more computational efficiency, therefore, of value in real implementations [19] [20] [21] [22] [23].

### 2.1. Visual-Based Slam

The accurate reconstruction of the captured scene from sets of ordered images has a long history in aerial [24] and close-range photogrammetry [25]. Usually, the object and reconstruction setup are well defined and the scene observations

using high-resolution cameras are well planed. Thus, the connectivity between multiple camera positions is known or easily established and off-line bundle adjustment (BA) over the cameras and scene structure is performed yielding accurate results. In addition, initial values for the exterior and interior camera orientations are mostly available from external sensors and accurate calibration.

### 2.2. Monocular Slam

In Monocular SLAM [26] [27] [28] [29] approach every frame is processed through the filter to mutually estimate the map landmarks locations and the camera pose. This approach has drawbacks of less computation efficiency in processing consecutive frames with any new information and the accumulation of linearization errors. While the keyframe-based approaches [30] [31] estimate the map using selected frames (keyframes) allowing performing accurate BA optimizations, as mapping is not tied to framerate. Strasdat *et al.* demonstrated that keyframe-based techniques are more accurate than filtering for the same computational efficiency [32]. The parallel tracking and mapping (PTAM) by Klein and Murray [31] introduce the idea of splitting camera tracking and mapping in parallel threads. The PTAM demonstrated to be successful for real time augmented reality applications in small environments [33]. The map points of PTAM correspond to Factored Solution to (FAST) SLAM corners matched by patch correlation. This makes the points only useful for tracking but not for place recognition. Also, PTAM does not detect large loops, and the re-localization is based on the correlation of low resolution of the keyframes, yielding a low invariance to viewpoint. Strasdat *et al.* [6] presented a large-scale monocular SLAM system with a front-end based on optical flow implemented on a GPU, followed by FAST feature matching and motion only BA, and a backend based on sliding-window BA [34].

### 2.3. Stereo Slam

Paz *et al.* [5] was the early stereo SLAM solution based on EKF-SLAM that was able to operate in larger environments than other approaches at that time. Most importantly, it was the first stereo SLAM exploiting both close and far points using an inverse depth parametrization [6] for the latter. They empirically showed that points can be reliably triangulated if their depth is less than 40 times the stereo baseline. Most modern stereo SLAM systems are keyframe-based [7] and perform BA optimization in a local area to achieve capability. The work of Strasdat *et al.* [8] performs a joint optimization of BA (point-pose constraints) in an inner window of keyframes and pose-graph (pose-pose constraints) in an outer window. By limiting the size of these windows, the method achieves constant time complexity, at the expense of not guaranteeing global consistency. The SLAM of Mei *et al.* [9] uses a relative representation of landmarks and poses and performs relative BA in an active area which can be constrained for constant time. This SLAM solution is able to close loops which allow expanding active

areas at both sides of a loop, but global consistency is not enforced. The recent SLAM by Pire *et al.* [10] performs local BA; however it lacks large loop closing. Similar to these approaches' BA is performed in a local set of keyframes so that the complexity is independent of the map size and can be operated in large environments. When closing a loop, the system aligns first both sides, similar to the SLAM solution in [9], so that the tracking is able to continue localizing using the old map and then performs a pose-graph optimization that minimizes the drift accumulated in the loop, followed by full BA. The recent Stereo LSD-SLAM of Engel *et al.* [11] is a semi-dense direct approach that minimizes photometric error in image regions with high gradient. Not relying on features, the method is expected to be more robust to motion blur or poorly textured environments.

## 3. Methodology

At the time the vehicle/rover moving the SLAM build a map of the surrounding environment and at the same time use this map to estimate its location with respect the landmarks. In SLAM both the trajectory of the vehicle/rover and the location of all landmarks are estimated without the need for any a priori knowledge of location. **Figure 1** shows the SLAM process in which the rover is moving in a specific environment and capture a set of landmarks, which is used to estimate the rover's location.

**Figure 1** shows thevehicle/rover measuring relative observations of some unknown landmarks with the SLAM sensor setup. The state vector $x_k$ describing the location and orientation of the vehicle/roverat time $k$, while the control vector $u_k$ is applied at time $k-1$ to move the vehicle/rover to a state $x_k$ at time $k$. $m_i$ is a vector describing the location of the $i^{\text{th}}$ landmark which true location is presumed time invariant. $z_{ik}$ is an observation measured from the vehicle/rover of the location of the $i^{\text{th}}$ landmark at time $k$. When there are several landmark observations at the same time, the observation will be expressed as $z_k$. The history of vehicle locations can be described as $X_{0:k} = \{x_0, x_1, \cdots, x_k\} = \{X_{0:k-1}, x_k\}$ and the history of control inputs is $U_{0:k} = \{u_1, u_2, \cdots, u_k\} = \{U_{0:k-1}, u_k\}$. In addition, the landmark location is $m = \{m_1, m_2, \cdots, m_n\}$ and the set of landmark observations can described as $Z_{0:k} = \{z_1, z_2, \cdots, z_k\} = \{Z_{0:k-1}, z_k\}$ [29] [30] [31].

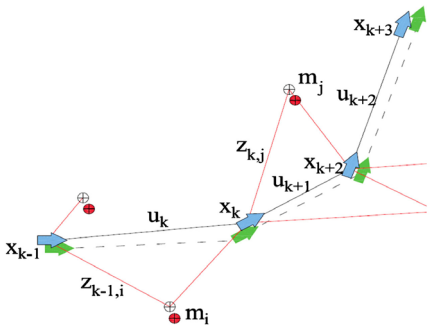



**Figure 1.** Arover moving in a specific environment using SLAM solution.

The Simultaneous Localisation and Mapping (SLAM) problem in probabilistic form demands that the probability distribution be calculated for all times $k$ as $P\left(x_k, m \mid Z_{0:k}, U_{0:k}, x_0\right)$. This probability distribution defines the joint posterior density of the landmark locations and vehicle/roverstate (at time $k$) given that the measured observations and control inputs including time $k$ along with the initial state of the vehicle/rover. Basically, a recursive solution to the SLAM problem is required. Beginning with an estimate for the distribution $P\left(x_{k-1}, m \mid Z_{0:k-1}, U_{0:k-1}\right)$ at time $k-1$, the joint posterior, following an observation $z_k$, and control $u_k$ is calculated. This calculation needs that a state transition model and an observation model are defined to describe the effect of the control input and observation respectively. The observation model defines the probability of taking an observation $z_k$ when the vehicle/roverlocation and landmark locations are known. The motion model for the vehicle/rover can be formulated as a probability distribution on state transitions in the form $P\left(x_k \mid x_{k-1}, u_k\right)$. Therefore, the state transition is assumed to be a Markov process in which the next state $x_k$ is independent of both the observations and the map, and depends only on the immediately proceeding state $x_{k-1}$ and the applied control $u_k$ [30].

The SLAM algorithm is currently implemented in a standard two-step recursive (sequential) prediction (time-update) as shown in Equation (1) and correction (measurement-update) form as shown in Equation (2)

$$\begin{aligned} &P\left(x_k, m \mid Z_{0:k-1}, U_{0:k}, x_0\right) \\ &= \int P\left(x_k \mid x_{k-1}, u_k\right) \times P\left(x_{k-1}, m \mid Z_{0:k-1}, U_{0:k-1}, x_0\right) \mathrm{d}x_{k-1} \end{aligned} \tag{1}$$

$$P\left(x_k, m \mid Z_{0:k}, U_{0:k}, x_0\right) = \frac{P\left(z_k \mid x_k, m\right) P\left(x_k, m \mid Z_{0:k-1}, U_{0:k}, x_0\right)}{P\left(z_k \mid Z_{0:k-1}, U_{0:k}\right))} \tag{2}$$

Equations (1) and (2) describe a recursive procedure for calculating the joint posterior $P\left(x_k, m \mid Z_{0:k}, U_{0:k}, x_0\right)$ for the vehicle/roverstate $x_k$ and map m at a time $k$ depend on all observations $Z_{0:k}$ and all control inputs $U_{0:k}$ including time $k$. The recursion is a function of a vehicle/rovermodel $P\left(x_k \mid x_{k-1}, u_k\right)$ and an observation model $P\left(z_k \mid x_k, m\right)$. In addition, the mapping problem could be formulated as calculating the conditional density $P\left(m \mid X_{0:k}, Z_{0:k}, U_{0:k}\right)$. This assumes that the location of the vehicle/rover $x_k$ is known at all different times, subject to known of initial location. Then, a map m is constructed by merging observations from different locations. On the other hand, the localisation problem may be defined as calculating the probability distribution $P\left(x_k \mid Z_{0:k}, U_{0:k}, m\right)$. This assumes that the landmark locations are known with certainty and the objective is to calculate an estimate of vehicle/roverlocation relative to these landmarks [30] [31].

## 4. Extended Kalman Filter Slam (EKF-SLAM)

In estimation theory, the extended Kalman filter (EKF) is the nonlinear version of the Kalman filter which linearizes about an estimate of the current mean and

covariance. The EKF-SLAM describe the vehicle/rover motion and observation model as shown in Equations (3) and (4) [28] [29].

$$P\left(x_k \mid x_{k-1}, u_k\right) \Leftrightarrow x_k = f\left(x_{k-1}, u_k\right) + w_k \tag{3}$$

where $f$ models vehicle kinematics and where $w_k$ are additive, zero mean uncorrelated Gaussian motion disturbances with covariance $Q_k$.

$$P\left(x_k \mid x_{k-1}, u_k\right) \Leftrightarrow z(k) = h\left(x_k, m\right) + v_k \tag{4}$$

where $h$ describes the geometry of the observation and where $v_k$ are additive, zero mean uncorrelated Gaussian observation errors with covariance $R_k$. With these definitions the standard EKF method can be applied to compute the mean and covariance of the joint posterior distribution $P\left(x_k, m \mid Z_{0:k}, U_{0:k}, x_0\right)$ as shown in Equations (5) and (6) with time-update in Equations (7) and (8) [29].

$$\begin{bmatrix} \hat{x}_{k|k} \\ \hat{m}_k \end{bmatrix} = E\begin{bmatrix} x_k \\ m \end{bmatrix} \mid Z_{0:k} \tag{5}$$

$$P_{k|k} = \begin{bmatrix} P_{xx} & P_{xm} \\ P_{xm}^T & P_{mm} \end{bmatrix}_{k|k} = E\left[\begin{pmatrix} x_k - \hat{x}_k \\ m - \hat{m}_k \end{pmatrix}\begin{pmatrix} x_k - \hat{x}_k \\ m - \hat{m}_k \end{pmatrix}^{\mathrm{T}} \mid Z_{0:k}\right] \tag{6}$$

$$\hat{x}_{k|k-1} = f\left(\hat{x}_{k-1|k-1}, u_k\right) \tag{7}$$

$$P_{xx,k|k-1} = \nabla f\, P_{xx,k-1|k-1} \nabla f^{\mathrm{T}} + Q_k \tag{8}$$

where $\nabla f$ is the Jacobina matrix of $f$ evaluated at the estimate $\hat{x}_{k-1|k-1}$. As the landmark are stationary, there will be no need for the time update. In addition, Equations (12) and (13) describe the observation update model [29] [30] [31].

$$\begin{bmatrix} \hat{x}_{k|k} \\ \hat{m}_k \end{bmatrix} = \begin{bmatrix} \hat{x}_{k|k-1} \\ \hat{m}_{k-1} \end{bmatrix} + W_k\left[z(k) - h\left(\hat{x}_{k|k-1}, \hat{m}_{k-1}\right)\right] \tag{9}$$

$$P_{k|k} = P_{k|k-1} - W_k\, S_k W_k^{\mathrm{T}} \tag{10}$$

where

$$S_k = \nabla h P_{k|k-1} \nabla h^{\mathrm{T}} + R_k \tag{11}$$

$$W_k = P_{k|k-1} \nabla h^{\mathrm{T}} S_k^{-1} \tag{12}$$

where $\nabla h$ is the Jacobian of h evaluated at $\hat{x}_{k|k-1}$ and $\hat{m}_{k-1}$.

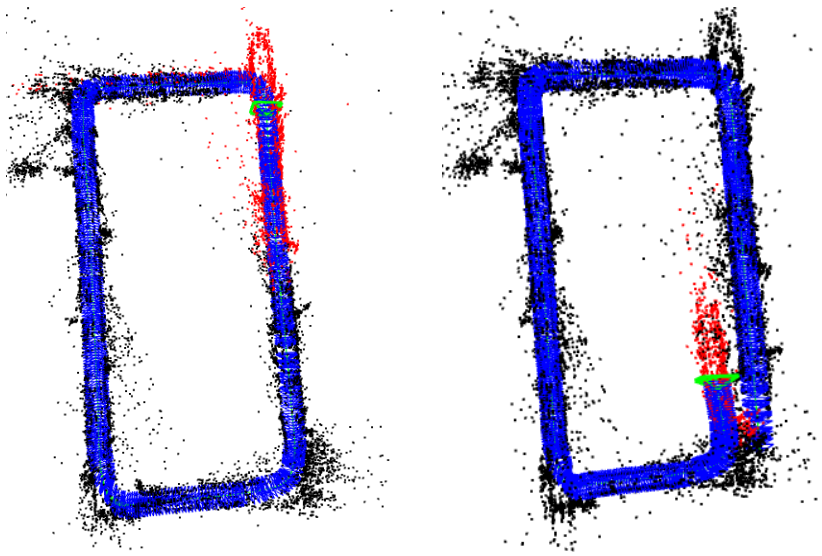
The loop-closure, when a vehicle/roverreturns to re-observe landmarks after a large traverse, is especially difficult. The association problem is compounded in environments where landmarks are not simple points and indeed look different from different viewpoints. EKF-SLAM employs linearized models of non-linear motion and observation models and so inherits many cautions. Non-linearity can be a significant problem in EKF-SLAM and leads to expected, and sometimes dramatic, inconsistency in solutions [24]. Convergence and consistency can only be guaranteed in the linear case [28]-[33].

## 5. Results and Discussions

All SLAM systems introduced in this paper have been implemented in different

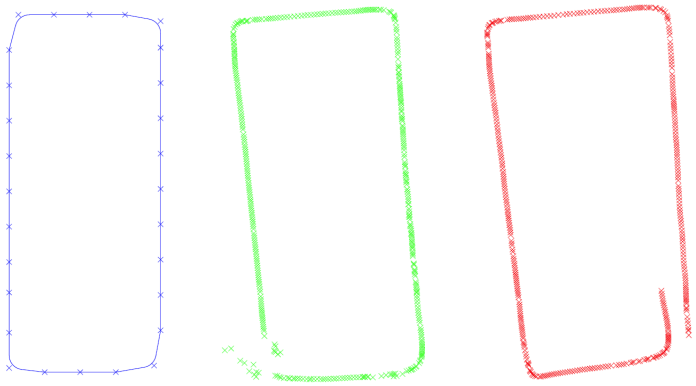
environment from indoor to outdoor. Several samples have been taken in different trajectory shape such as a straight line that enable us to measure the drift in addition to the loop sample that we use it to test the closure and its corresponding trajectory deformation. A ground reference has been captured for all the samples to measure the biases and drift caused by the SLAM solution. Figure 2 shows the results of the monocular loop SLAM solution in indoor environment. The trajectory and the surrounding captured environment for the indoor loop with closure and without closure are shown in Figure 2.

The results shown in Figure 2 show that the monocular SLAM solution has a drift that cause non closure loop as shown in Figure 1(a). The indoor loop monocular SLAM trajectories without closure and with closure are compared to the indoor loop ground truth in Figure 3. The ground truth is very close to a rectangular shape while the Monocular SLAM trajectory suffered from different drifts, mostly because the turns, which results in a skewed open rectangular shape (Figure 3(c)). However, this drift has been adjusted by applying the loop closure algorithm constraints. Loop closures were solved with a pose graph optimization with similarity constraints, that was able to correct the scale drift appearing in monocular SLAM. However, the loop closure constraints location estimation causes an additional deformation for the loop with respect to the ground truth, as shown in Figure 3(b). In order to adjust the monocular scale drift, a well-known object has been used to be used as reference to adjust the scale along the trajectory. This procedure produces an initialization drift because the camera will be targeted to the ground before the vehicle/rover start to move along the trajectory. After few seconds the camera will be rotated to face the moving direction of the trajectory. In order to produce a precise mapping solution, this procedure requires a post processing correction.



(a) (b)

Figure 2. Visual monocular SLAM solution for indoor environment with loop closure (a) and without loop closure (b).



(a) (b) (c)

**Figure 3.** A comparison between the indoor loop trajectory reference (a) and the visual monocular SLAM trajectory with loop closure (b) and without loop closure (c).

The second sample for the indoor monocular SLAM solution was a straight line. **Figure 4** shows the results for the indoor monocular straight-line SLAM trajectory and surrounding environment (**Figure 4(b)** and **Figure 4(c)**) and the ground truth (**Figure 4(a)**) which extends to 118.90 m.

**Figure 4** shows that the monocular straight-line SLAM suffered from a drift of 3.36% of the total line length. The results of the monocular SLAM show a low-density mapping for the surrounding environment. Similar to monocular SLAM, stereo SLAM samples have been captured in both loop and line scenarios. The results of the stereo loop SLAM trajectory and surrounding environment for the indoor loop with closure and without closure are shown in **Figure 5(a)** and **Figure 5(b)**, respectively. In case of stereo SLAM without loop closure, the loop SLAM solution didn't close itself due to accumulated drift the however this drift is less than the monocular SLAM situation. **Figure 5(a)** shows the stereo SLAM loop with closure constraints.

**Figure 6** shows a comparison between the indoor loop stereo SLAM trajectories without closure and with closure with the indoor loop ground truth. The drift caused by the stereo SLAM solution is less than the drift monocular SLAM as shown in **Figure 6**. In addition, the loop closure deformation effect is less than the deformation generated by the monocular SLAM as the stereo loop has rectangular shape after the closure and close to the ground truth, as shown in **Figure 6**.

**Figure 7** shows the results of the stereo indoor SLAM straight line trajectory with the surrounding environment and the ground truth. It is shown that the drift of the stereo trajectory has a drift of 1.85% of the total line length. The results of the stereo SLAM solution show a higher-density mapping for the surrounding environment than the monocular SLAM.

On the other hand, outdoor SLAM solutions have been introduced including monocular, and stereo visual sensors. **Figure 8** shows the results for the monocular SLAM solution for the outdoor environment with and without loop closure. Due to the wide field of view, there are more captured feature than the

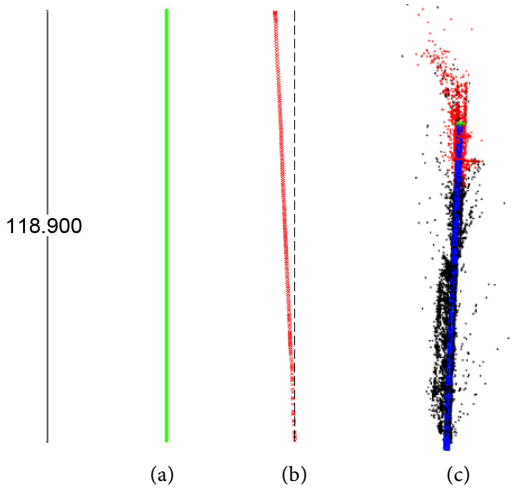



**Figure 4.** A comparison between the indoor straight-line trajectory ground truth (a) and the visual monocular line SLAM trajectory (b) and the captured trajectory with the surrounding environment (c).

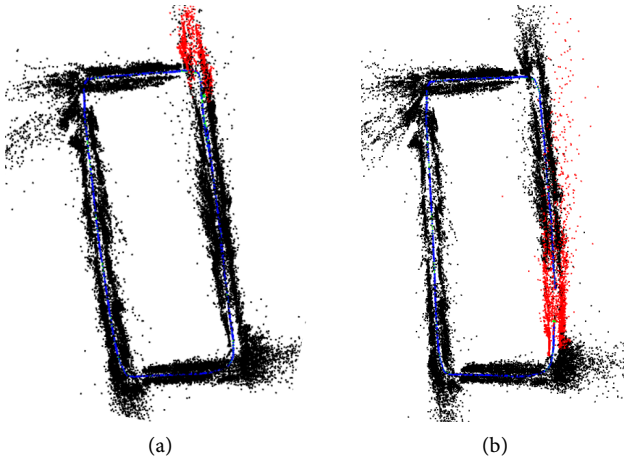



**Figure 5.** Visual Stereo SLAM solution for indoor environment with loop closure (a) and without loop closure (b).

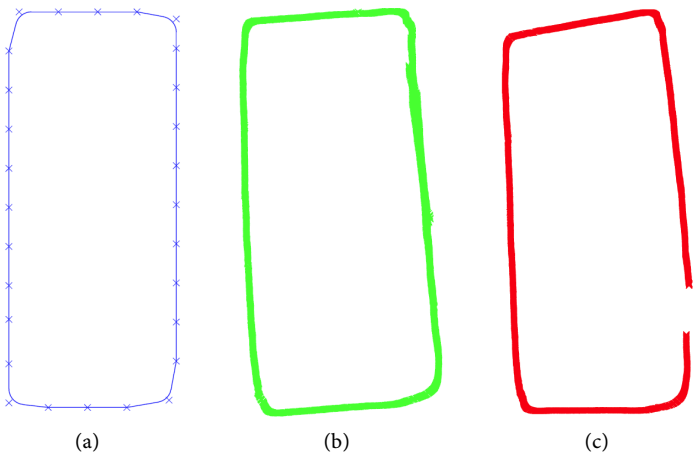



**Figure 6.** A comparison between the indoor loop trajectory reference (a) and the visual stereo SLAM trajectory with loop closure (b) and without loop closure (c).

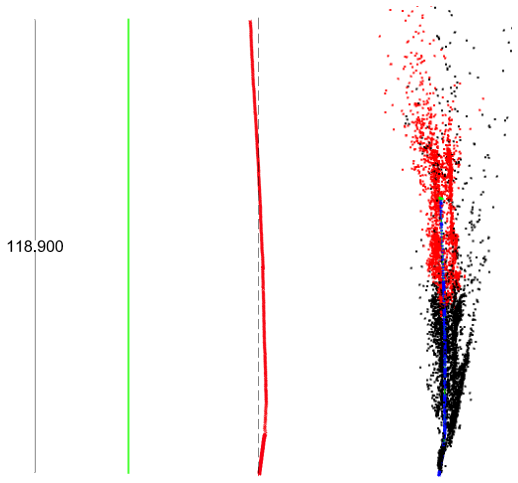



Figure 7. A comparison between the indoor straight-line trajectory reference (a) and the visual stereo line SLAM trajectory (b) and the captured trajectory with the surrounding environment (c).

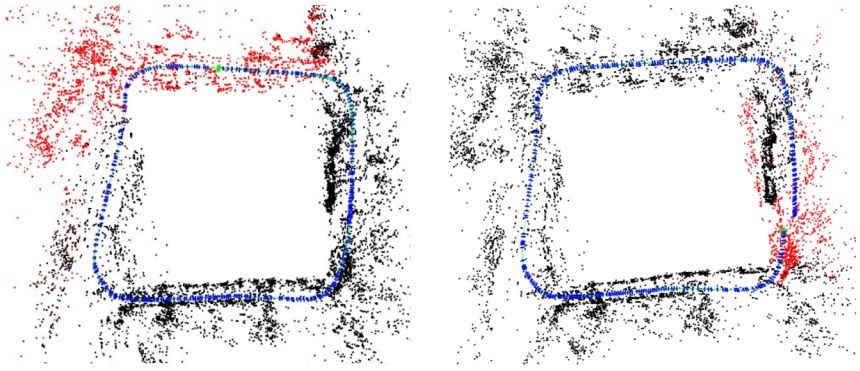
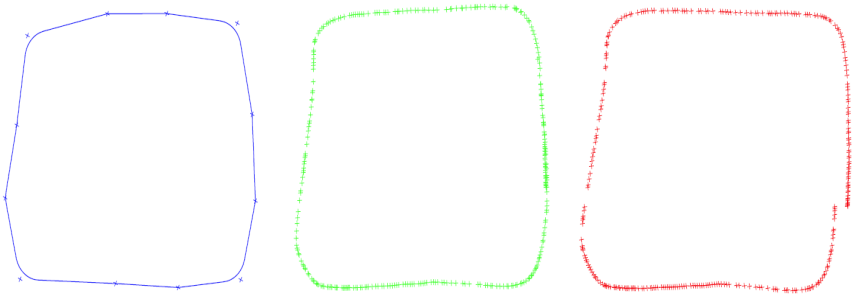



Figure 8. Visual monocular SLAM solution for outdoor environment with loop closure (a) and without loop closure (b).

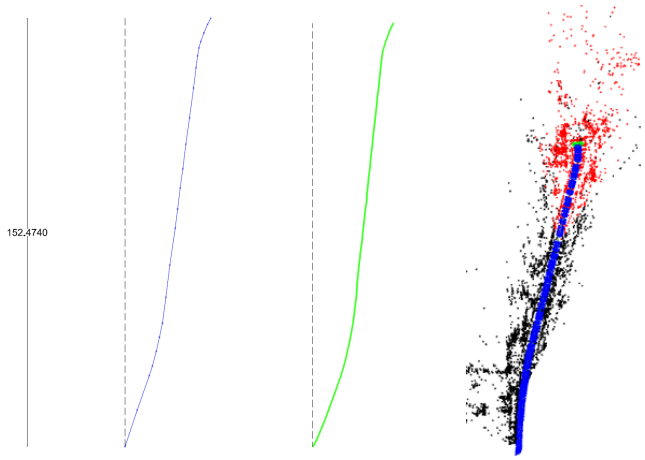
indoor environments. However, both indoor and outdoor have the same map density. Similar to the indoor monocular SLAM solution, the outdoor solution suffers from drifts that cause open loop. However, applying the loop closure constraints results in a closed loop with less deformation than the indoor loop. There is less loop deformation, which may be due to the wide field of view with more well distributed captured features. Figure 9 shows the Stereo SLAM trajectories for outdoor environment with loop closure (Figure 9(b)) and without loop closure (Figure 9(c)) compared to the outdoor ground truth loop (Figure 9(a)). The outdoor SLAM straight line is captured with several curves along the line to test the drift.

Figure 10 shows the results of the monocular SLAM line trajectory and surrounding environment compared to the ground truth. It is shown that the monocular SLAM trajectory has a drift of 10% of the total line length. The outdoor drift of the line is higher than the indoor line as the outdoor includes curves.



(a) (b) (c)

Figure 9. A comparison between the outdoor loop trajectory reference (a) and the visual monocular SLAM trajectory with loop closure (b) and without loop closure (c).




(a) (b) (c)

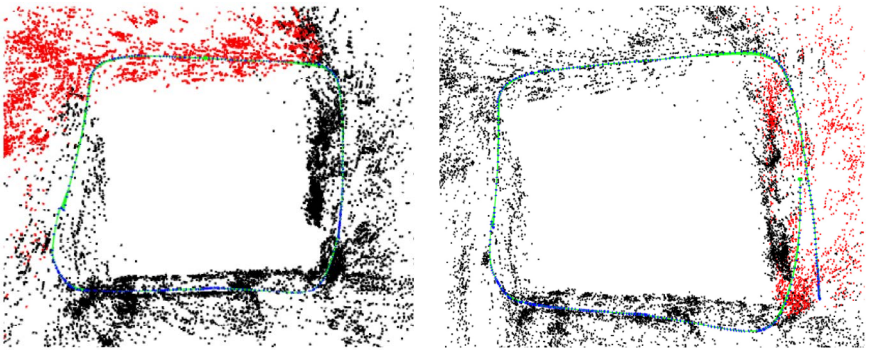
Figure 10. A comparison between the outdoor line trajectory reference (a) and the visual monocular line SLAM trajectory (b) and the captured trajectory with the surrounding environment (c).

The outdoor stereo SLAM results are shown in Figure 11 including the trajectory with and without loop closure and the surrounding environment. It is shown that the captured map has higher density than the indoor stereo SLAM map, due to the wider field of view.

The stereo outdoor SLAM solution results are shown in Figure 12 including the trajectory with and without loop closure compared to the ground truth. The loop deformation caused by loop closure constraints in the outdoor stereo SLAM is less than the indoor stereo SLAM loop deformation.

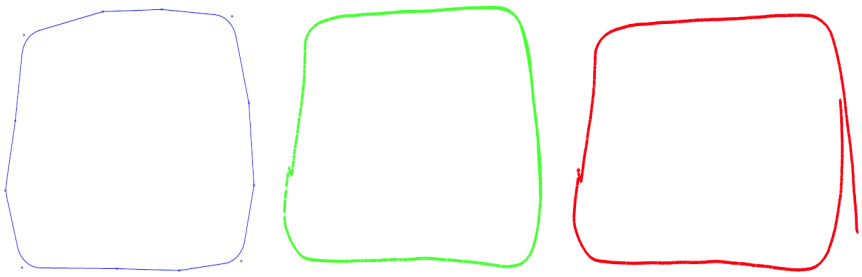
In addition, Figure 13 shows the outdoor results for the stereo line SLAM trajectory compared to the ground truth with the surrounding environment. The stereo line SLAM solution has a drift of 1.5% of the total line length.

The horizontal root mean squares error (RMSE) of the SLAM trajectories for Monocular, and stereo solutions are shown in Figure 14. As shown in the results, the indoor results of the monocular SLAM are more precise than the outdoor. However, the outdoor results of the stereo SLAM are more precise than the indoor results by 30%, which a result of the small stereo baseline cameras.



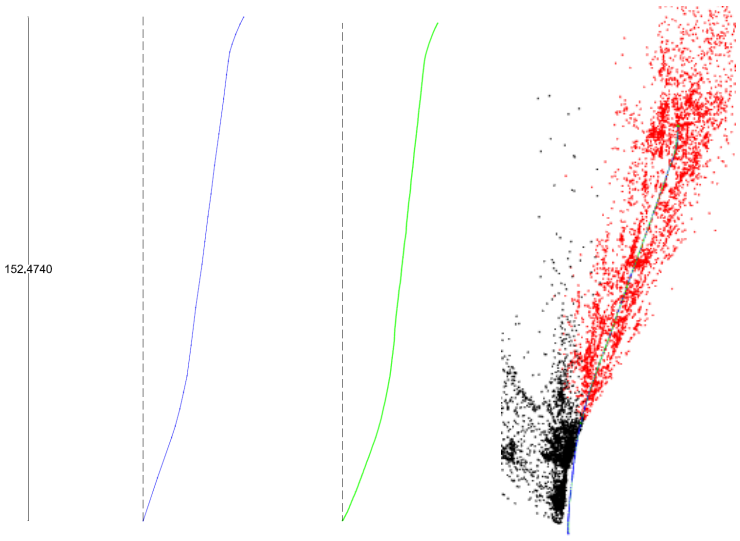
(a) (b)

Figure 11. Visual Stereo SLAM solution for outdoor environment with loop closure (a) and without loop closure (b).



(a) (b) (c)

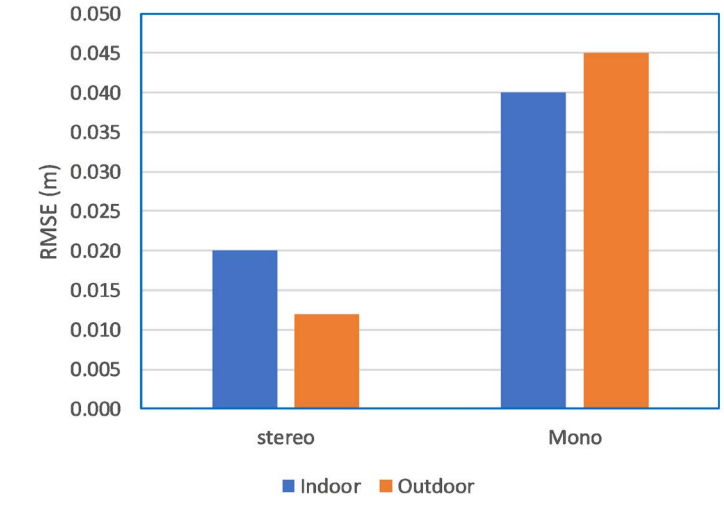
Figure 12. A comparison between the outdoor loop trajectory reference (a) and the visual stereo SLAM trajectory with loop closure (b) and without loop closure (c).



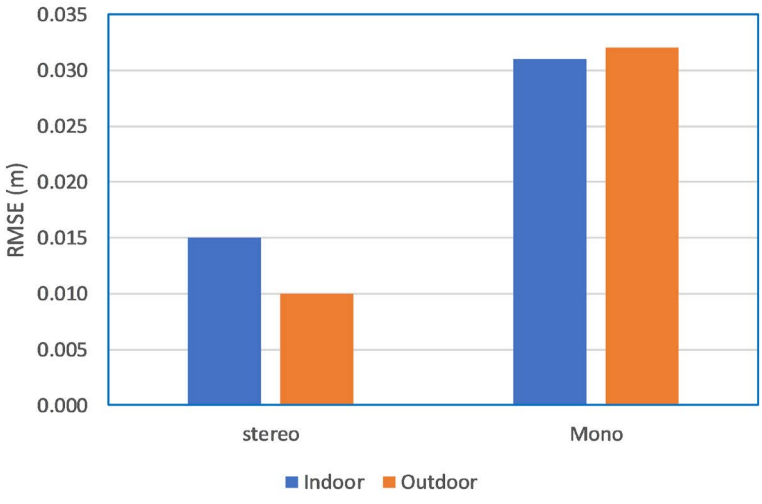

(a) (b) (c)

Figure 13. A comparison between the outdoor straight-line trajectory reference (a) and the visual stereo line SLAM trajectory (b).

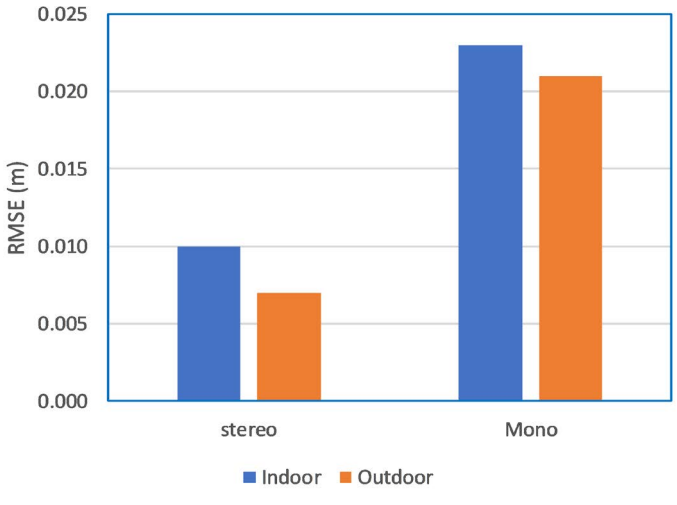
Figures 16-21 show the error in the vertical localization SLAM results with the angular rotation around an $x$ axis, perpendicular to the direction of movement, as shown in Figure 15.




(a)




(b)




(c)

**Figure 14.** The horizontal localization RMSE for the indoor and outdoor SLAM trajectory using Monocular, and Stereo of straight line (a), loop without closure (b), and loop with closure (c).

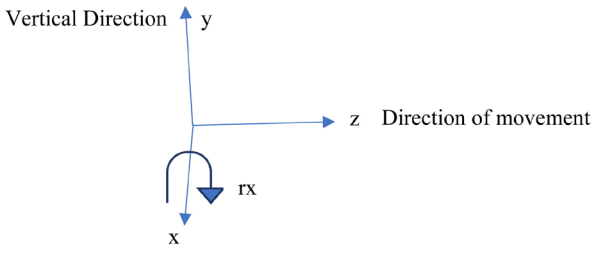
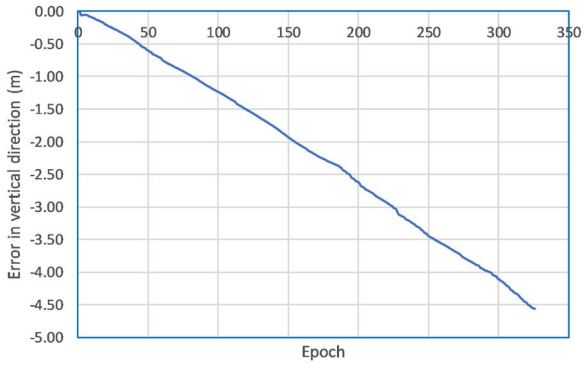
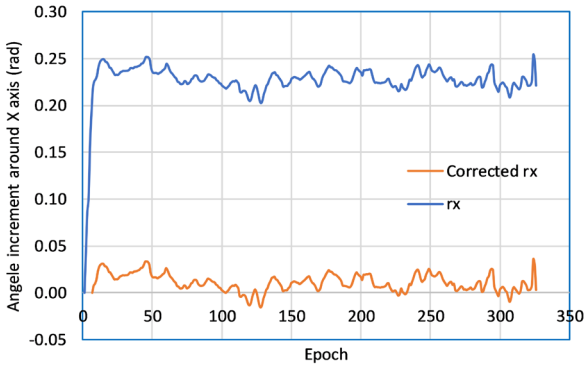



**Figure 15.** SLAM orientation axis and the direction of movement.



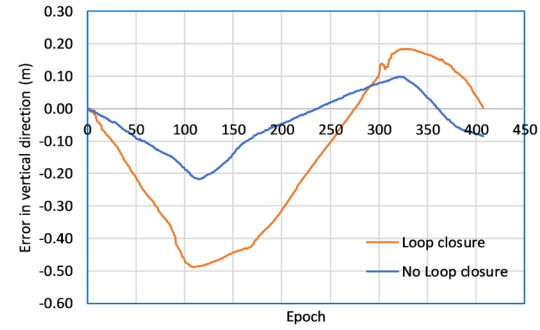

(a)



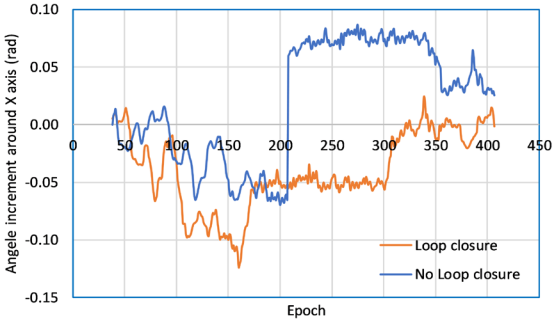

(b)

**Figure 16.** The vertical position error for the monocular line SLAM trajectory (a), and the angular rotation around *x*-axis (b).

Figure 16 and Figure 17 show the vertical monocular localization error in in both the straight line and loop trajectories SLAM with the angular rotation around *x*-axis. It is shown that the angular rotation has a jump in the beginning which related to the monocular SLAM setup initialization which the camera was initially facing downwards towards the ground before being rotated to face the direction of movement. However, if we ignore the first few epochs of the trajectory the angular rotation will be corrected. On the other hand, the error in the vertical direction of the line increases in a linear pattern, which reach to 3.8% of the total trajectory length. In the loop SLAM results, the vertical error is less than the line as the trajectory turns the rotation angle changes its direction. However, applying the loop closure constraints doesn't improve the overall precision.

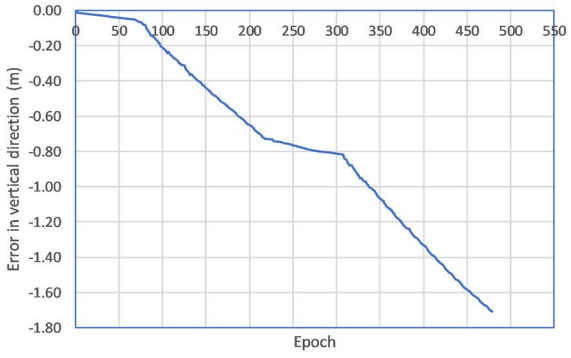



(a)

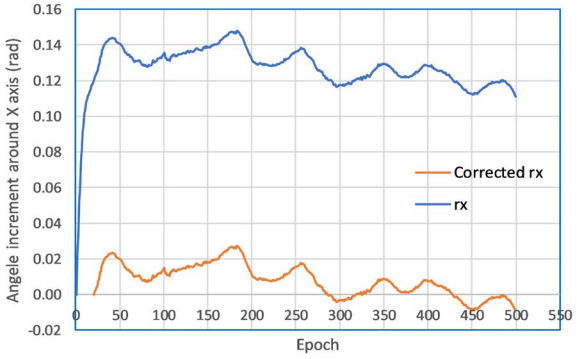



(b)

**Figure 17.** The vertical position error for the monocular loop SLAM trajectory (a), and the angular rotation around *x*-axis (b).

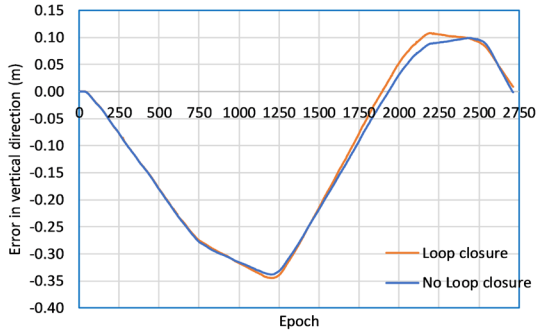



(a)

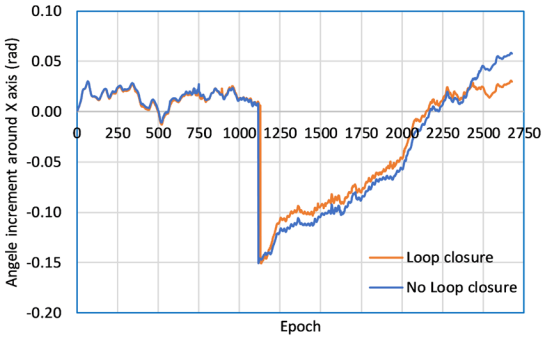



(b)

**Figure 18.** The vertical position error for the stereo line SLAM trajectory (a), and the angular rotation around *x*-axis (b).

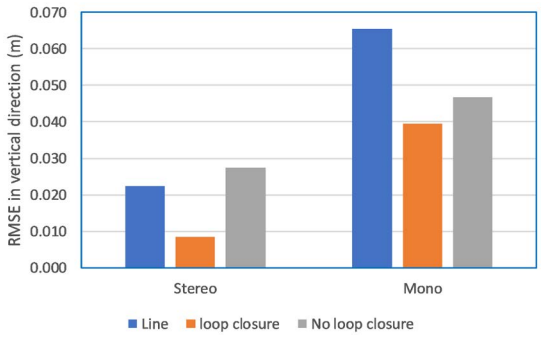



(a)

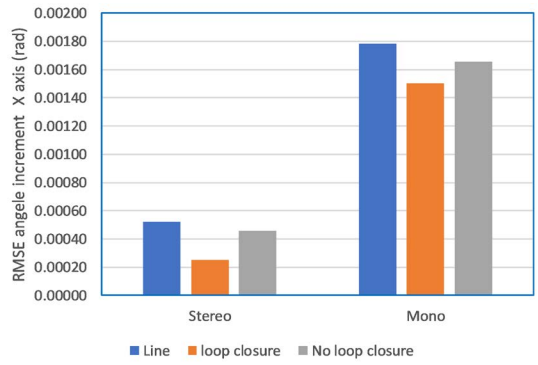



(b)

**Figure 19.** The vertical position error for the stereo loop SLAM trajectory (a), and the angular rotation around *x*-axis (b).

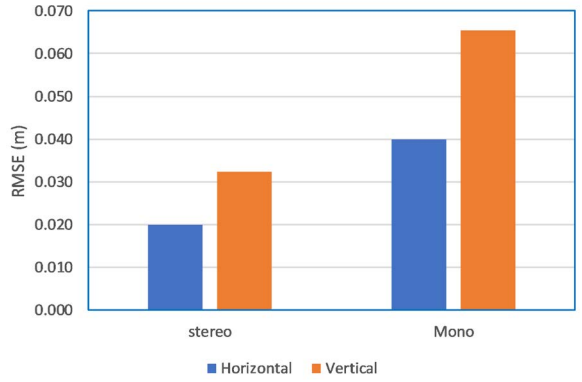



(a)

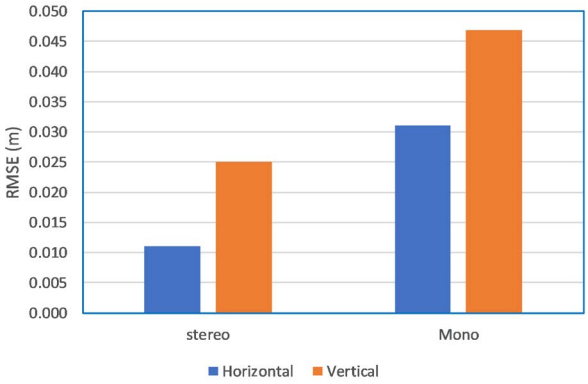



(b)

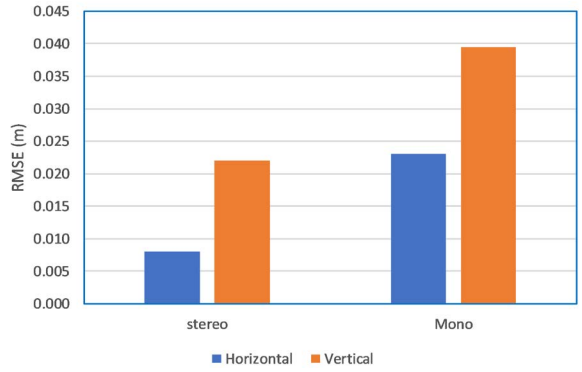
**Figure 20.** The RMSE for the vertical SLAM trajectory using monocular, and stereo of straight line (a), and the angular rotation around *x*-axis (b).




(a)




(b)




(c)

**Figure 21.** The RMSE positioning for the horizontal and vertical SLAM trajectory using monocular, and stereo of straight line (a), loop without closure (b), and loop with closure.

The vertical SLAM localization results are shown in **Figure 18** and **Figure 19** for both line and loop trajectories, respectively. It is shown that the error in the stereo vertical localization line SLAM is 1.4% of the total trajectory which is almost 50% less than the monocular SLAM line results. The angular rotation in the stereo SLAM loop suffers from a jump in the beginning similar to the monocular SLAM system as both system's setup initialization causes this rotation. On the other hand, the error in the vertical loop SLAM localization is less than the results of the monocular SLAM results.

**Figure 20** shows the RMSE for the vertical localization component of mono-

cular, and stereo SLAM solutions for the line, loop closure, and loop without closure. It is shown that loop closure of the stereo SLAM solutions has the higher vertical localization precision.

A comparison between RMSE of the horizontal and vertical localization components are shown in Figure 21. Generally, the horizontal localization results are more precise than the vertical results, as shown in Figure 21.

## 6. Conclusion

In this paper, SLAM system is introduced using monocular and stereo visual sensors. SLAM solutions are implemented in both indoor and outdoor. The SLAM samples have been taken in different modes such as a straight line that enable us to measure the drift in addition to the loop sample that is used to test the closure and its corresponding trajectory deformation. In order to verify the trajectory scale, a baseline method has been used. In addition, a ground truth has been captured for all the samples indoor and outdoor to measure the biases and drifts caused by the SLAM solution. Both monocular and stereo SLAM data have been captured with the same visual sensors which in the stereo situation had a baseline of 20.00 cm. It has been shown that, the stereo SLAM localization results are 75% higher precision than the monocular SLAM solution. In addition, the indoor results of the monocular SLAM are more precise than the outdoor. However, the outdoor results of the stereo SLAM are more precise than the indoor results by 30%, which is a result of the small stereo baseline cameras. In the vertical SLAM localization component, the stereo SLAM generally shows 60% higher precision than the monocular SLAM results.

## Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

## References

[1] Mur-Artal, R. and Tardos, J.D. (2014) ORB-SLAM: Tracking and Mapping Recognizable Features. *Proceedings of Robotics*: *Science and Systems*, Berkeley, USA.

[2] Mur-Artal, R. and Tardos, J.D. (2015) Probabilistic Semi-Dense Mapping from Highly Accurate Feature-Based Monocular SLAM. *Proceedings of Robotics*: *Science and Systems*, Rome, Italy. https://doi.org/10.15607/RSS.2015.XI.041

[3] Mur-Artal, R. and Tardos, J.D. (2016) ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. *IEEE Transactions on Robotics*, **33**, 1255-1262. https://doi.org/10.1109/TRO.2017.2705103

[4] Mur-Artal, R., Montiel, J.M.M. and Tardos, J.D. (2015) ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Transactions on Robotics*, **31**, 1147-1163. https://doi.org/10.1109/TRO.2015.2463671

[5] Kerl, C., Sturm, J. and Cremers, D. (2013) Dense Visual SLAM for RGB-D Cameras. *Conference Intelligent Robots and Systems*, November 2013, 2100-2106. https://doi.org/10.1109/IROS.2013.6696650

[6] Fioraio, N. and Konolige, K. (2011) Real Time Visual and Point Cloud Slam. *Pro-*

*ceedings of the RGB-D Workshop on Advanced Reasoning with Depth Cameras at Robotics*: *Science and Systems*, Los Angeles, USA.

[7] Hess, W., Kohler, D., Rapp, H. and Andor, D. (2016) Real-Time Loop Closure in 2D LIDAR SLAM. *IEEE International Conference on Robotics and Automation*, Montreal, 20-24 May 2019, 1271-1278. https://doi.org/10.1109/ICRA.2016.7487258

[8] Zhang, J. and Singh, S. (2014) LOAM: Lidar Odometry and Mapping in Real-Time. *Robotics*: *Science and Systems Conference*, Berkeley, July 2014. https://doi.org/10.15607/RSS.2014.X.007

[9] Berkeley Localization and Mapping. https://github.com/erik-nelson/blam

[10] Andújar, D., Escolà, A., Rosell-Polo, J.R., Fernández-Quintanilla, C. and Dorado, J. (2013) Potential of a Terrestrial LiDAR-Based System to Characterise Weed Vegetation in Maize Crops. *Computers and Electronics in Agriculture*, **92**, 11-15. https://doi.org/10.1016/j.compag.2012.12.012

[11] Ehlert, D. and Heisig, M. (2013) Sources of Angle-Dependent Errors in Terrestrial Laser Scanner-Based Crop Stand Measurement. *Computers and Electronics in Agriculture*, **93**, 10-16. https://doi.org/10.1016/j.compag.2013.01.002

[12] Hosoi, F. and Omasa, K. (2009) Estimating Vertical Plant Area Density Profile and Growth Parameters of a Wheat Canopy at Different Growth Stages Using Three Dimensional Portable Lidar Imaging. *ISPRS Journal of Photogrammetry and Remote Sensing*, **64**, 151-158. https://doi.org/10.1016/j.isprsjprs.2008.09.003

[13] Hosoi, F. and Omasa, K. (2009) Estimation of Vertical Plant Area Density Profiles in a Rice Canopy at Different Growth Stages by High-Resolution Portable Scanning LIDAR with a Lightweight Mirror. *ISPRS Journal of Photogrammetry and Remote Sensing*, **74**, 11-19. https://doi.org/10.1016/j.isprsjprs.2012.08.001

[14] Osterman, A., Godeša, T., Hocevar, M., Širok, B. and Stopar, M. (2013) Real-Time Positioning Algorithm for Variable-Geometry Air-Assisted Orchard Sprayer. *Computers and Electronics in Agriculture*, **98**, 175-182. https://doi.org/10.1016/j.compag.2013.08.013

[15] Rosell-Polo, J.R., Llorens, J., Sanz-Cortiella, R., Arnó-Satorra, J., Ribes-Dasi, M., Masip, J., Escolà, A., Camp, F., Solanelles-Batlle, F., Gràcia, F., Gil, E., Val, L., Planas-Demartí, S. and Palacin-Roca, J. (2009) Obtaining the Three-Dimensional Structure of Tree Orchards from Remote 2D Terrestrial LIDAR Scanning. *Agricultural and Forest Meteorology*, **149**, 1505-1515. https://doi.org/10.1016/j.agrformet.2009.04.008

[16] Saeys, W., Lenaerts, B., Craessaerts, G. and De Baerdemaeker, J. (2009) Estimation of the Crop Density of Small Grains Using LiDAR Sensors. *Biosystems Engineering*, **102**, 22-30. https://doi.org/10.1016/j.biosystemseng.2008.10.003

[17] Weiss, U. and Biber, P. (2011) Plant Detection and Mapping for Agricultural Robots Using a 3D LIDAR Sensor. *Robotics and Autonomous Systems*, **59**, 265-273. https://doi.org/10.1016/j.robot.2011.02.011

[18] Côté, J.F., Widlowski, J.L., Fournier, R.A. and Verstraete, M.M. (2009) The Structural and Radiative Consistency of Three-Dimensional Tree Reconstructions from Terrestrial Lidar. *Remote Sensing of Environment*, **113**, 1067-1081. https://doi.org/10.1016/j.rse.2009.01.017

[19] Keightley, K.E. and Bawden, G.W. (2010) 3D Volumetric Modeling of Grapevine Biomass Using Tripod LiDAR. *Computers and Electronics in Agriculture*, **74**, 305-312. https://doi.org/10.1016/j.compag.2010.09.005

[20] Rosell-Polo, J.R., Sanz-Cortiella, R., Llorens, J., Arnó-Satorra, J., Escolà, A., Ribes-Dasi, M., Masip, J., Camp, F., Gràcia, F., Solanelles-Batlle, F., Pallejà-Cabré,

T., Val, L., Planas-Demartí, S., Gil, E. and Palacin-Roca, J. (2009) A Tractor-Mounted Scanning LIDAR for the Non-Destructive Measurement of Vegetative Volume and Surface Area of Tree-Row Plantations: A Comparison with Conventional Destructive Measurements. *Biosystems Engineering*, **102**, 128-134. https://doi.org/10.1016/j.biosystemseng.2008.10.009

[21] Sanz-Cortiella, R., Rosell-Polo, J.R., Llorens, J., Gil, E. and Planas-Demartí, S. (2013) Relationship between Tree Row LIDAR-Volume and Leaf Area Density for Fruit Orchards and Vineyards Obtained with a LIDAR 3D Dynamic Measurement System. *Agricultural and Forest Meteorology*, **171**-**172**, 153-162. https://doi.org/10.1016/j.agrformet.2012.11.013

[22] Durrant-Whyte, H. and Bailey, T. (2006) Simultaneous Localization and Mapping: Part I. *IEEE Robotics & Automation Magazine*, **13**, 99-110. https://doi.org/10.1109/MRA.2006.1638022

[23] Bailey, T. and Durrant-Whyte, H. (2006) Simultaneous Localization and Mapping (SLAM): Part II. *IEEE Robotics &Automation Magazine*, **13**, 108-117. https://doi.org/10.1109/MRA.2006.1678144

[24] Kneip, L., Siegwart, R. and Pollefeys, M. (2012) Finding the Exact Rotation between Two Images Independently of the Translation. *Proceedings of the European Conference on Computer Vision*, Florence, October 2012, 696-709. https://doi.org/10.1007/978-3-642-33783-3_50

[25] Luhmann, T., Robson, S., Kyle, S. and Harley, I. (2006) Close Range Photogrammetry: Principles, Methods and Applications. Whittles, Dunbeath, 528.

[26] Davison, A.J., Reid, I.D., Molton, N.D. and Stasse, O. (2007) MonoSLAM: Real-Time Single Camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **29**, 1052-1067. https://doi.org/10.1109/TPAMI.2007.1049

[27] Civera, J., Davison, A.J. and Montiel, J.M.M. (2008) Inverse Depth Parametrization for Monocular SLAM. *IEEE Transactions on Robotics*, **24**, 932-945. https://doi.org/10.1109/TRO.2008.2003276

[28] Chiuso, A., Favaro, P., Jin, H. and Soatto, S. (2002) Structure from Motion Causally Integrated over Time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**, 523-535. https://doi.org/10.1109/34.993559

[29] Eade, E. and Drummond, T. (2006) Scalable Monocular SLAM. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 1, New York, June 2006, 469.

[30] Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F. and Sayd, P. (2006) Real Time Localization and 3d Reconstruction. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 1, 363-370. https://doi.org/10.1109/CVPR.2006.236

[31] Klein, G. and Murray, D. (2007) Parallel Tracking and Mapping for Small AR Workspaces. *IEEE and ACM International Symposium on Mixed and Augmented Reality*, Nara, November 2007, 225-234. https://doi.org/10.1109/ISMAR.2007.4538852

[32] Strasdat, H., Montiel, J.M.M. and Davison, A.J. (2012) Visual SLAM: Whyfilter? *Image and Vision Computing*, **30**, 65-77. https://doi.org/10.1016/j.imavis.2012.02.009

[33] Klein, G. and Murray, D. (2008) Improving the Agility of Keyframe-Based SLAM. *European Conference on Computer Vision*, Marseille, October 2008, 802-815. https://doi.org/10.1007/978-3-540-88688-4_59

[34] Portugal, D., Araújo, A. and Couceiro, M.S. (2019) A Guide for 3D Mapping with Low-Cost Sensors Using ROS. In: Koubaa, A., Ed., *Robot Operating System* (*ROS*), Springer, Berlin, 3-23. https://doi.org/10.1007/978-3-030-20190-6_1