Experimental Verification of Direct Depth Computing Technique for Monocular Visual SLAM Systems

Mohamed Hasan
Mechatronics and Robotics Engineering
Egypt-Japan University of Science and Technology
Alexandria, Egypt
mohamed.hasan@ejust.edu.eg

Mohamed Abdellatif
Mechatronics and Robotics Engineering
Egypt-Japan University of Science and Technology
Alexandria, Egypt
mohamed.abdellatif@ejust.edu.eg

Abstract—**this paper verifies a recently published method of monocular depth computing in the context of visual SLAM. The closed form depth solution was exploited in the measurement model of a monocular EKF visual SLAM algorithm. SIFT interest points are tracked during camera motion and a suitable feature initialization is presented. The visual SLAM system is verified through experiments on a mobile robot platform and the results are benchmarked to groundtruth.**

Keywords-monocular depth from motion; monocular visual SLAM ; SIFT.

I. INTRODUCTION

Simultaneous localization and mapping, SLAM is an important problem for several robotics applications [1]. Using a camera as the main sensor, Visual SLAM, VSLAM has attracted many researchers as a leading area of interest [2-4]. Depth calculation is necessary for this process and both mono [5-6] and stereo [7-8] cameras have been used with VSLAM.

Computing monocular depth has been introduced through estimation [9], closed-form solution [10] and learning [11] techniques. For the applications of mobile robots, depth recovery from visual information is necessary for navigation. The estimation techniques of depth from motion take a number of iterations until settled to a depth value and thus affecting both the speed and accuracy of robotic missions like obstacle avoidance, localization and mapping. On the contrary, closed-form solutions can compute the depth in real time without iterations and thus enable fast and safe navigation of the mobile robots.

One of the recent methods of depth calculation for VSLAM is the inverse depth parameterization, IDP [12]. Although IDP had been successfully exploited for monocular VSLAM, it still has two drawbacks. Firstly, the initial uncertainty around features position is infinite and remains so until the depth solution settles. Secondly, IDP has been calculated relative to image frame of reference not the world frame. Although this has the merit of decreasing the uncertainty, it lacks the ability of working with online SLAM. That is because the results of the VSLAM system using such IDP can be transformed from image to world domain using the groundtruth data [13]. This second drawback is very harmful for mobile robot applications.

A new closed-form solution for depth from motion was presented in [14]. The solution recovers the depth of a static point in real time without the need of iterative estimation. Both the camera pose and the calibration parameters information have been used to compute the depth. The new depth solution builds on the notation that any interest point existing in an image of a static scene, has a static location in the world regardless of camera motion (with the implicit assumption of static world features). This constraint is used to calculate the depth of an image point from just two monocular views of that point [14]. The requirements and results of this depth solution are suitable for monocular VSLAM in terms of computational speed and accuracy.

In this paper, the recently published closed form solution of depth [14] is verified in the context of EKF mono VSLAM. The depth solution will be exploited in the measurement model of the VSALM system and proper feature initialization is presented. Experiments have been made on an indoor mobile robot platform carrying a webcam. Naturally appearing feature points are extracted using SIFT method and tracked during camera motion. Results are benchmarked with measured groundtruth robot motion.

The paper is organized as follows. The closed-form solution is introduced in the next section. The computational framework for EKF monocular VSLAM system is described in section 3. Results of real experiments on an indoor mobile robot are presented and discussed in section 4. Finally, conclusions are given in section 5.

II. CLOSED FORM DEPTH SOLUTION

The depth solution of [14] works under the following explicit assumptions:

- The camera is calibrated for intrinsic parameters.
- The observed features are static.
- The initial robot pose is known.

The calibrated camera is assumed to move and a sequence of monocular images is captured. Consider a static point featur $p_i^w = (X \quad Y \quad Z)^T$ represented in world coordinates. The camera observes p_i^w twice from two different world

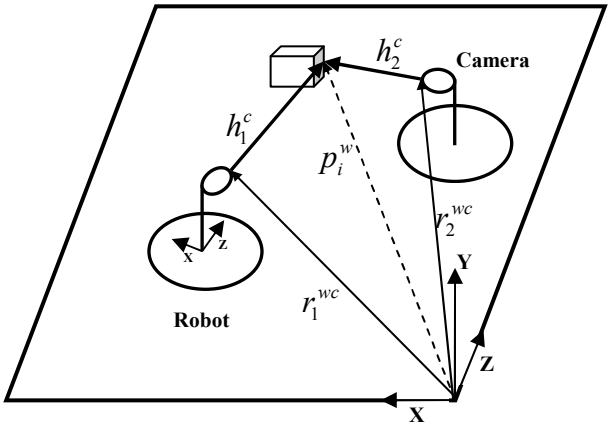


Figure 1. The mobile robot with the onboard camera observing a feature from two different positions

positions: r_1^{WC}, r_2^{WC} as shown in Fig. 1. The distances between the camera and p_i^w in the two observations are denoted as h_1^c and h_2^c which are represented in the camera coordinates. Transformation from camera to world coordinates is performed by the rotation matrices: R_1^{WC}, R_2^{WC} which are functions of the camera pitch angles: θ_1, θ_2. The pitch angle is the rotation of the planar robot around the vertical axis Y^C. The projections of p_i^w on the image plane are (u_1, v_1) in the first image and (u_2, v_2) in the other image.

The intrinsic parameters of the camera are known from prior camera calibration. Standard pinhole camera model- without lens distortion- has been used along with these intrinsic parameters. The distance between the camera and an observed point is composed of three Euclidean components; $h^c = (h_x \quad h_y \quad h_z)^T$. Thus, the pixel coordinates of the observed point are given by the following camera model:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u_0 - \dfrac{f\,h_x}{d\,h_z} \\ v_0 - \dfrac{f\,h_y}{d\,h_z} \end{pmatrix} \tag{1}$$

where u_0, v_0 are the camera center in pixels, f is the focal length and d is the pixel size.

The world location of the point feature can be determined given the location of the camera and the distance from this location to the point. This may be written for the two different observations as:

$$p_i^W = r_1^{WC} + R_1^{WC} h_1^C \tag{2}$$

$$p_i^W = r_2^{WC} + R_2^{WC} h_2^C \tag{3}$$

where:

$$r_i^{WC} = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix},\ R_i^{WC} = \begin{pmatrix} \cos\theta_i & 0 & \sin\theta_i \\ 0 & 1 & 0 \\ -\sin\theta_i & 0 & \cos\theta_i \end{pmatrix} \tag{4}$$

From (1):

$$h_i^c = \begin{pmatrix} h_{xi} \\ h_{yi} \\ h_{zi} \end{pmatrix} = \begin{pmatrix} \dfrac{d}{f}(u_0 - u_i)h_{zi} \\ \dfrac{d}{f}(v_0 - v_i)h_{zi} \\ h_{zi} \end{pmatrix} = h_{zi} \begin{pmatrix} a_i \\ b_i \\ 1 \end{pmatrix} \tag{5}$$

where

$$a_i = \frac{d}{f}(u_0 - u_i),\ b_i = \frac{d}{f}(v_0 - v_i) \tag{6}$$

and i=1, 2 in (4-6).

It can be noted that the observed feature has a fixed world location regardless of camera motion. This constraint has been used to provide solution by equating (2), (3), using (4-6) and rearranging:

$$\begin{pmatrix} a_1\cos\theta_1 + \sin\theta_1 & -a_2\cos\theta_2 - \sin\theta_2 \\ b_1 & -b_2 \\ -a_1\sin\theta_1 + \cos\theta_1 & a_2\sin\theta_2 - \cos\theta_2 \end{pmatrix} \begin{pmatrix} h_{z1} \\ h_{z2} \end{pmatrix} = \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix} \tag{7}$$

or equivalently:

$$A\,h_z = B \tag{8}$$

where $h_z = (h_{z1} \quad h_{z2})^T$.

The depth of the feature can now be calculated from (8) using:

$$h_z = (A^T A)^{-1} A^T B \tag{9}$$

This completes the solution of the depth of the point.

III. EKF MONOCULAR VSLAM

The VSLAM problem is solved using the traditional EKF filter in the common prediction and update stages. The features are the SIFT [15] interest points detected as in [16]. The depth method presented in section II is selected to express depth from features to camera. The constant velocity model is used to describe the robot motion. The robot state is defined as:

$$x_r^w = \begin{pmatrix} z & x & \theta \end{pmatrix}^T \quad (10)$$

where z , x denote the robot position relative to world coordinates and θ is the pitch angle of robot rotation around Y^W . The state of any point feature in the map is given as:

$$f_i^w = \begin{pmatrix} x_f & y_f & z_f \end{pmatrix}^T \quad (11)$$

The full state vector comprises the robot state and the entire n map features:

$$Y = \begin{pmatrix} x_r^T, f_1^T, ..., f_n^T \end{pmatrix}^T \quad (12)$$

The mobile robot platform (Festo *Robotino*) used in experiments has three omnidirectional wheels. The exact robot's motion model controlling the EKF prediction stage is:

$$f_v = \begin{pmatrix} z_{k+1}^w \\ x_{k+1}^w \\ \theta_{k+1}^w \end{pmatrix} = \begin{pmatrix} z_k^w \\ x_k^w \\ \theta_k^w \end{pmatrix} + \Delta_t \begin{pmatrix} v_z \cos\theta - v_x \sin\theta \\ v_z \sin\theta + v_x \cos\theta \\ \omega \end{pmatrix} \quad (13)$$

where v_z , v_x and ω are the linear and angular robot velocities comprising the control vector u . The required EKF Jacobians are completed by differentiating $\left(\partial f_r / \partial x_r\right)$ and $\left(\partial f_r / \partial u\right)$.

A new feature is firstly measured in the image as $\left(u_1, v_1\right)$ and then initialized in the map using either (2) or (3):

$$f^w = r^{wc} + R^{wc} h^c \quad (14)$$

where the distance from camera, h^c is defined by (5) and the position of the camera is given by:

$$r^w = \begin{pmatrix} x & y_c & z \end{pmatrix}^T \quad (15)$$

where x , z are the robot's position and y_c is the fixed camera height. At initialization, the depth h_z is set to a suitable initial value which has been taken empirically as 1m in the given experiments. The covariance of the new feature requires the differentiation of $\left(\partial f^w / \partial x_r\right)$ and $\left(\partial f^w / \partial h_c\right)$.

Measurement prediction after robot motion is done by:

$$\tilde{h}_c = \tilde{R}_c^w \left(f_i^w - \tilde{r}^{wc} \right) \quad (16)$$

where $\tilde{R}_c^w$ and $\tilde{r}^{wc}$ are given by the robot predicted state $\tilde{x}_r$ computed after motion by EKF prediction. Note that this prediction is computed in terms of the initial depth of the added new feature. The EKF Jacobians for measurement require the differentiation of $\left(\partial h_c / \partial Y\right)$.

Once the added feature has a match in the captured image sequence, this match is measured as $\left(u_2, v_2\right)$ and the robot pose is acquired. This information is fed into (9) to compute the depth h_z of this feature. After that the new distance h^c after matching is calculated from (5) and recorded as the match z_c . The difference between the estimated and the matched feature distances $\left(z_c - \tilde{h}_c\right)$ is the innovation used in the EKF update stage. This means that any new feature is initialized at a constant initial depth (1m) and this depth is corrected after robot motion and observing the same feature again.

IV. Experiments and Results

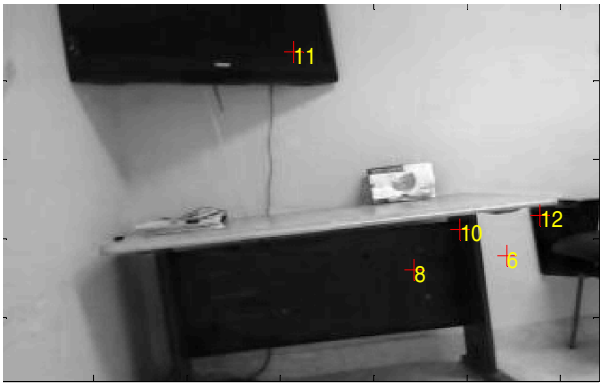
The monocular VSLAM system described in the previous sections is verified using a planar mobile robot with a webcam. A marker is fixed to the robot base and hanged freely in order to record the groundtruth of robot motion on the floor. Programmed control velocities were given to the robot to achieve squared motion path. The output of the VSLAM system is benchmarked to the recorded groundtruth motion. The robot was moved inside our lab space on two squared paths starting at different robot orientations.

In the first path, the robot has started form zero pitch angle, moved for 600 cm starting with forward motion and 77 images were captured. The tenth image from the captured sequence is shown in Fig. 2.a with the matched SIFT features marked by their registered numbers in the map. The XZ-map generated up to this moment is shown in Fig. 2.b with all the added features at that time. The robot is also shown as a circle representing the robot's base and a line denoting the orientation. The output path from the VSLAM system is shown in Fig.3 along with the groundtruth path. The path output from the EKF VSLAM is shown to estimate the robot real path well. The robot X and Z paths are plotted in Fig. 4 with the groundtruth against the captured frames. Although there was a deviation in the *Z* estimate at the 55th frame, the VSLAM system recovers again after re-observing the old features stored in map. The errors between the VSLAM localization output and the groundtruth are shown in Fig. 5. Error in *X* motion has not exceeded ± 10 cm, whereas the error in *Z* motion has reached larger values but it has converged at the end of the path and returned near the true end point.

In the second path, the initial robot orientation has been -150°, it moved for 480 cm starting with forward and captured 60 images. Similar to the previous paths, the results of this final path are presented in Fig. 6 to Fig.9. Statistics of the localization results of the three paths are given in Table 1.

Table 1. Localization errors of the three paths.

Path	Min error [cm]			Max error [cm]			Mean error [cm]		
	X	Z	XZ	X	Z	XZ	X	Z	XZ
First	0	0	0	12	41	42	5.1	7.6	9.9
Second	0	0	0	25	17	26	11	4.6	12



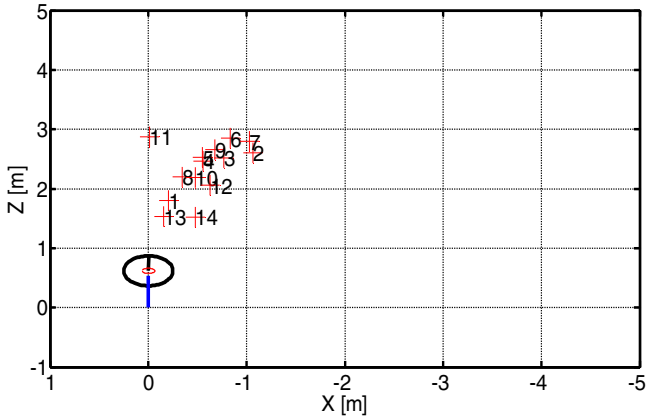
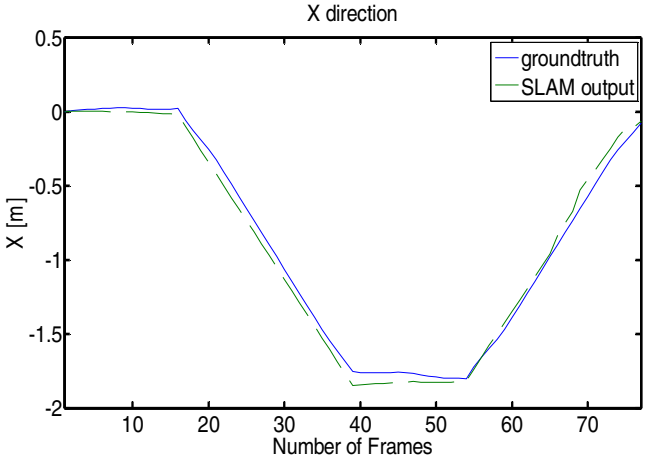


Figure 2. (a) The tenth image of first path and the matched SIFT features. (b) The generated map up to that time with the robot inside it shown as a circle.



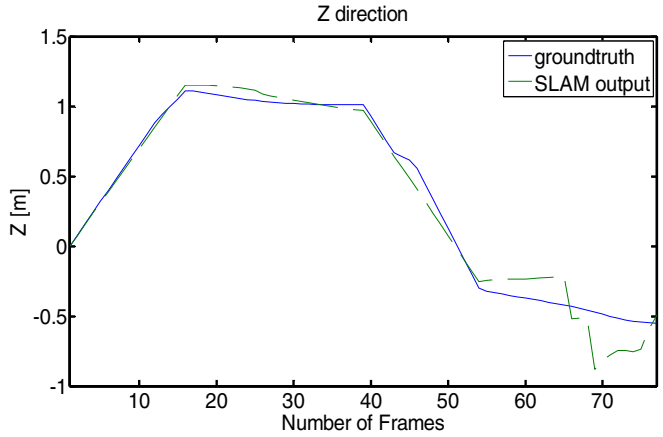


Figure 4. The X- and Z- VSLAM output and groundtruth of the first path.

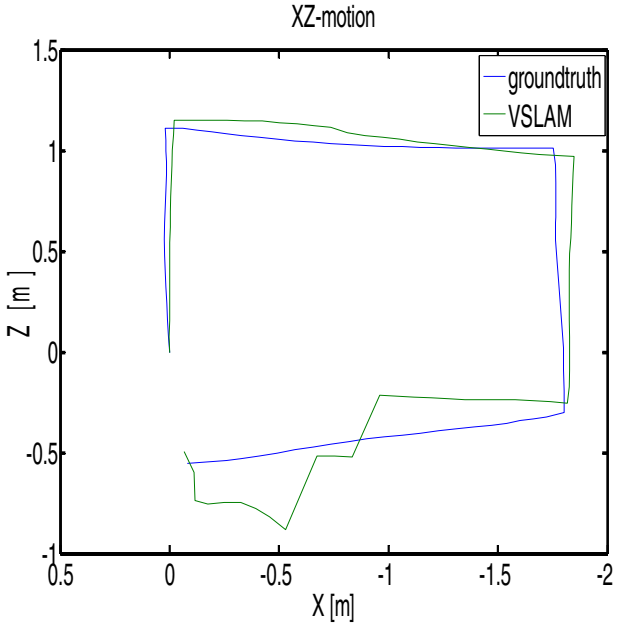


Figure 3. The XZ path output from VSLAM and the groundtruth of the first path.

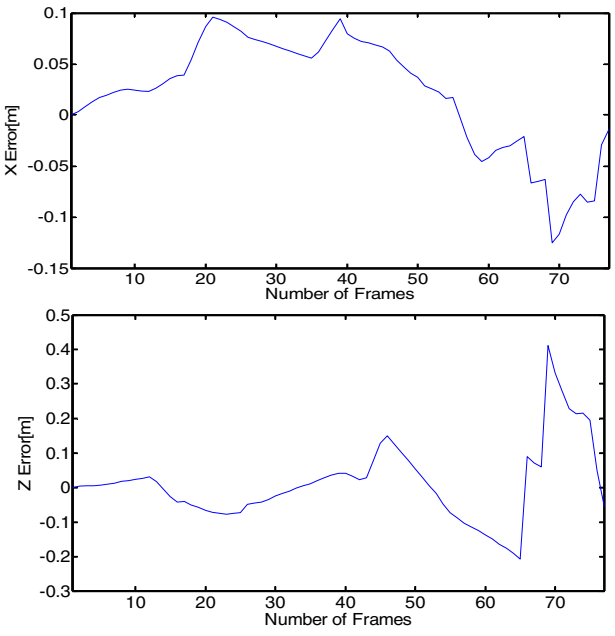


Figure 5. Errors in Robot localization in the first path.

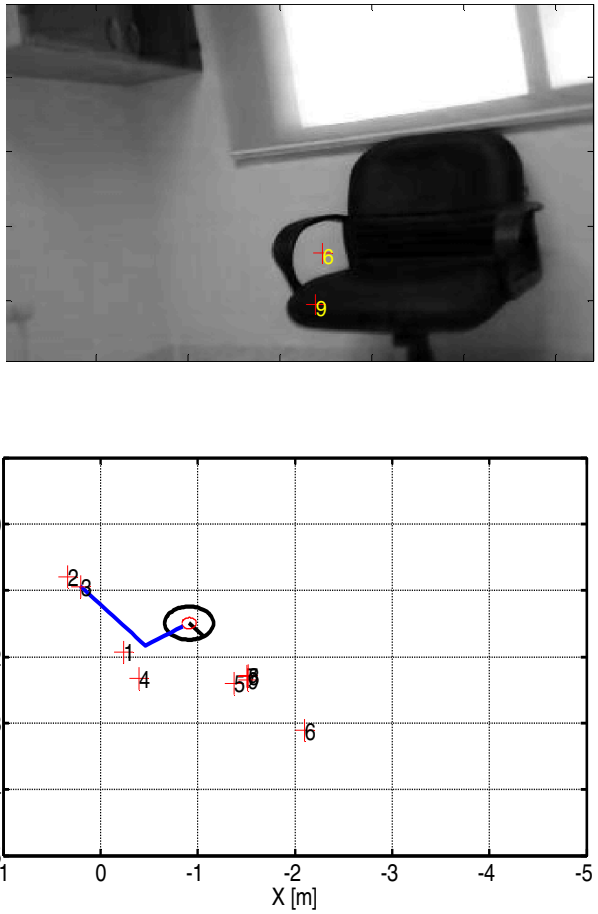


Figure 6. (a) A sample imag of the second path and the matched SIFT features. (b) The generated map up to that time with the robot inside it shown as a circle.

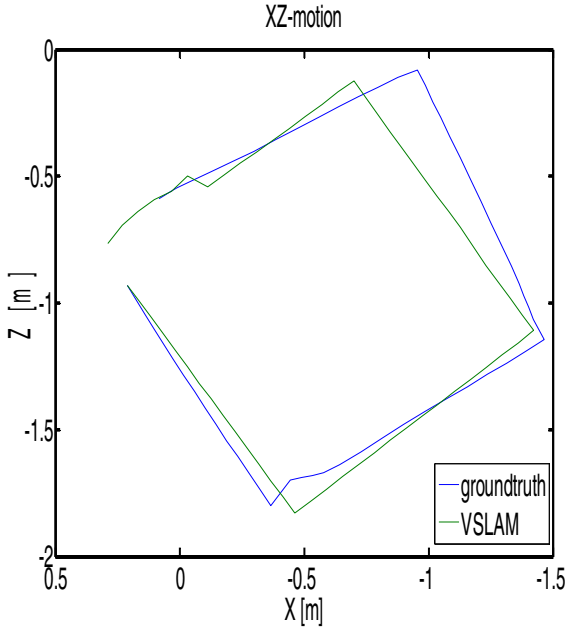


Figure 7. The output of VSLAM in the second path along with its

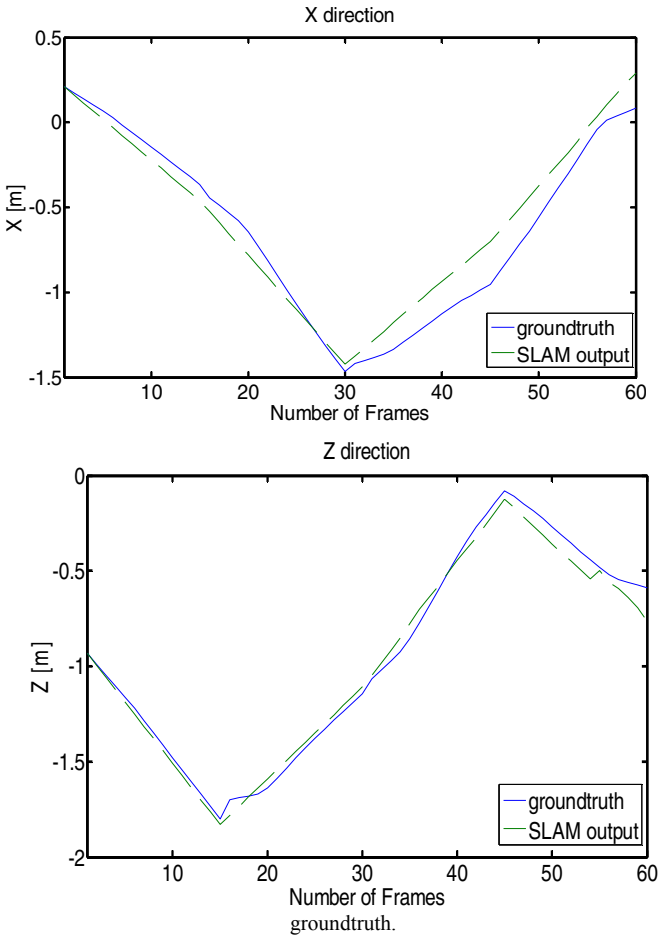


Figure 8. The X- and Z- VSLAM output and groundtruth of the second path.

groundtruth.

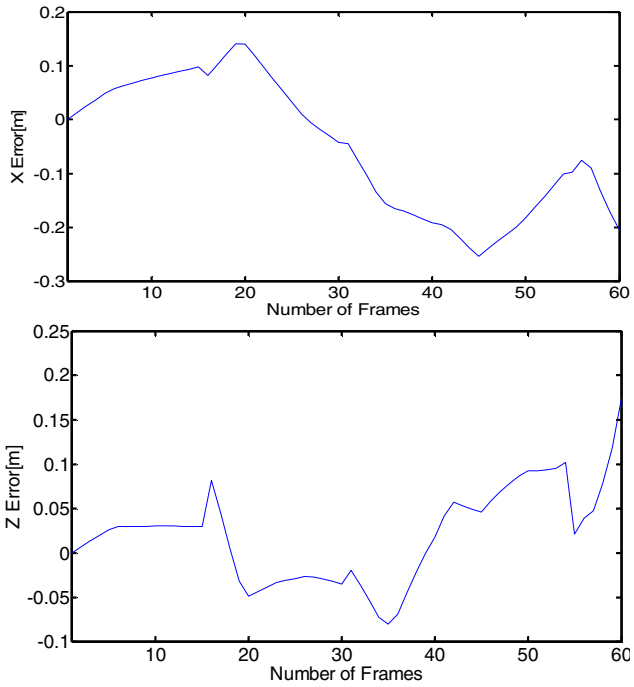


Figure 9. Locoalization errors in the third path.

V. CONCLUSIONS

A closed form solution of monocular depth from motion was presented in the context of visual SLAM. The depth computing method was used in the measurement model of an EKF monocular VSLAM system. Features can be initialized without a delay once they have been observed. An initial depth is assumed for the initialized feature, which will then be corrected upon matching of this feature after camera motion. This VSLAM system is verified in real experiments using a mobile robot. The results show the validity of the depth solution to be used in mono SLAM. One problem of the presented work is the large localization error that occurs in case of large camera orientation or matching errors. It is believed that the error can be further reduced by more careful considerations which will be part of our future research agenda.

ACKNOWLEDGMENT

The first author is supported by a scholarship from the Ministry of Higher Education, Government of Egypt which is gratefully acknowledged.

REFERENCES

[1] H. Durrant-Whyte and T. Bailey, “Simultaneous Localization and Mapping: Part I, ” IEEE Robotics and Automation Magazine, vol 13(2):pp. 99-110, 2006.

[2] A. Gil, Ó. Reinoso, M. Ballesta, M. Juliá, “Multi-robot visual SLAM using a Rao-Blackwellized particle filter, ” Robotics and Autonomous Systems, vol 58, pp. 68-80, 2010.

[3] G. Silveira, E. Malis, P. Rives, “An Efficient Direct Approach to Visual SLAM, ” IEEE Transactions On Robotics, vol 24(5), pp. 969-979, 2008.

[4] D. Zou and P. Tan, “CoSLAM: Collaborative Visual SLAM in Dynamic Environments, ” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012.

[5] M. Milford and G. Wyeth, “Mapping a Suburb With a Single Camera Using a Biologically Inspired SLAM System, ” IEEE Transactions On Robotics, vol 24(5), pp. 1038-1053, 2008.

[6] A. Davison, I. Reid, N. Molton, O. Stasse, “MonoSLAM: Real-Time Single Camera SLAM, “ IEEE Transactions on Pattern Analysis and Machine intelligence, vol 29(6), pp. 1052 - 1067, 2007.

[7] L. Paz, P. Pini, J. Tard, “Large-Scale 6-DOF SLAM With Stereo-in-Hand, “ IEEE Transactions On Robotics, vol 24(5), pp. 946-957, 2008.

[8] D. Schleicher,L. Bergasa, M. Ocaña, R. Barea, M. López,” Real-Time Hierarchical Outdoor SLAM Based on Stereovision and GPS Fusion, “ IEEE Transactions on Intelligent Transportation Systems, vol 10(3), pp. 440-452, 2009.

[9] U. Franke and C. Rabe, "Kalman filter based depth from motion with fast convergence, " IEEE Symp. on Intelligent Vehicles, pp. 181 - 186, 2005.

[10] Y. Murphey, J. Chen, J. Crossman, J. Zhang, P. Richardson and L. Sieh, " DepthFinder, A Real-time Depth Detection System for Aided Driving, " IEEE Symp. on Intelligent Vehicles, pp. 122-127, 2000.

[11] A. Saxena, S. Chung, A. Ng, "3-D Depth Reconstruction from a Single Still Image, " Int. J. of Computer Vision, Vol. 76, NO.1, pp. 53–69, 2008.

[12] J. Civera, A. Davison and J. Montiel, "Inverse Depth Parametrization for Monocular SLAM, " IEEE Trans. on Robotics, Vol. 24, NO.5, pp. 932–945, 2008.

[13] J. Civera, Ó. Grasa, A. Davison and J. Montiel, “1-Point RANSAC for EKF Filtering: Application to Real-Time Structure from Motion and Visual Odometry, ” Journal of Field Robotics, vol 27(5), pp. 609-631, 2010.

[14] M. Hasan and M. Abdellatif, “Monocular Depth from Motion Using a New Closed-Form Solution, ” Lecture Notes on Artificial Intelligenece, LNAI 7508, pp. 473–483, 2012.

[15] D. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints, ” International J. of Computer Vision, vol 60 (2), pp. 91-110, 2004.

[16] A. Vedaldi, B. Fulkerson, “{VLFeat}: An Open and Portable Library of Computer Vision Algorithms” , http://www.vlfeat.org, 2008.