Scientific Research Publishing

Positioning, 2020, 11, 45-60
https://www.scirp.org/journal/pos
ISSN Online: 2150-8526
ISSN Print: 2150-850X

Monocular VO Scale Ambiguity Resolution Using an Ultra Low-Cost Spike Rangefinder

Ahmed El Amin, Ahmed El-Rabbany

Department of Civil Engineering, Ryerson University, Toronto, Canada
Email: agelamin@Ryerson.ca, rabbany@ryerson.ca

How to cite this paper: El Amin, A. and El-Rabbany, A. (2020) Monocular VO Scale Ambiguity Resolution Using an Ultra Low-Cost Spike Rangefinder. *Positioning*, **11**, 45-60.
https://doi.org/10.4236/pos.2020.114004

Received: November 4, 2020
Accepted: November 27, 2020
Published: November 30, 2020

Abstract

Monocular visual odometry (VO) is the process of determining a user's trajectory through a series of consecutive images taken by a single camera. A major problem that affects the accuracy of monocular visual odometry, however, is the scale ambiguity. This research proposes an innovative augmentation technique, which resolves the scale ambiguity problem of monocular visual odometry. The proposed technique augments the camera images with range measurements taken by an ultra-low-cost laser device known as the Spike. The size of the Spike laser rangefinder is small and can be mounted on a smartphone. Two datasets were collected along precisely surveyed tracks, both outdoor and indoor, to assess the effectiveness of the proposed technique. The coordinates of both tracks were determined using a total station to serve as a ground truth. In order to calibrate the smartphone's camera, seven images of a checkerboard were taken from different positions and angles and then processed using a MATLAB-based camera calibration toolbox. Subsequently, the speeded-up robust features (SURF) method was used for image feature detection and matching. The random sample consensus (RANSAC) algorithm was then used to remove the outliers in the matched points between the sequential images. The relative orientation and translation between the frames were computed and then scaled using the spike measurements in order to obtain the scaled trajectory. Subsequently, the obtained scaled trajectory was used to construct the surrounding scene using the structure from motion (SfM) technique. Finally, both of the computed camera trajectory and the constructed scene were compared with ground truth. It is shown that the proposed technique allows for achieving centimeter-level accuracy in monocular VO scale recovery, which in turn leads to an enhanced mapping accuracy.

Keywords

Spike, Visual Odometry, Monocular, Scale

1. Introduction

Visual odometry (VO) is a process, which estimates the camera poses from a series of successive images [1] [2] [3]. The process, which is considered as a core element of simultaneous localization and mapping (SLAM), has been widely used in the field of robotics [4] [5] [6]. VO is capable of determining accurate trajectories when stereo cameras are used [7]. This is due mainly to the availability of accurate and consistent depth information, which can then be used to determine the scale value. However, a stereo camera pair will potentially degenerate into a monocular camera when the baseline between the stereo camera becomes much smaller than the distance between camera and scene [8]. Additionally, the stereo visual odometry system requires self-calibration after a long-term operation to reduce the mechanical vibration encountered in the implementation [1] [9]. In contrast, monocular cameras are compact and inexpensive compared with stereo cameras. However, monocular VO suffers from scale ambiguity. The monocular camera cannot compute the length of translational movement from feature correspondences only, as the distance between the camera and the features cannot be estimated by triangulation directly.

Scale ambiguity can be retrieved by imposing additional information, including known initials, additional constraints, and the addition of other sensors. Klein and Murray [10] assumed that the translation length of the initial camera movement is known, while Davison *et al.* [6] started a visual mono SLAM from a predefined landmark. However, the use of initial assumptions leads to a scale drift as a result of error accumulation over time.

Additional constraints, such as known camera height above the ground, have also been proposed to resolve scale ambiguity. Kitt *et al.* [11] and Choi *et al.* [12] extracted the ground surface and applied it to calculate the planar homography matrix to retrieve the scale factor. Song *et al.* [13] used a combination of several cues within a predefined region, which allowed for highly accurate ground plane estimation and consequently led to a scale recovery. Nevertheless, these approaches are constrained by limited information about environments. Gakne *et al.* [14] presented a method for estimating the scale ambiguity and reduce drift by using a 3D building model. The correction of the scale improved the positioning solution by 90% compared to a solution that does not correct the scale drift. However, these methods need an available 3D model with known precision.

The addition of other sensors was considered by some researchers to resolve the VO scale ambiguity, either through direct or indirect measurement. As an example, Scaramuzza *et al.* [15] adopted a speedometer as an additional sensor, where the scale factor is computed through the time difference between the camera frames and the corresponding vehicle speed. Unfortunately, however, it is not always possible to use a speedometer. A low-cost inertial measurement unit (IMU) was used by [16] [17] to estimate the scale factor in VO. Unfortunately, low-cost IMUs suffer from significant error accumulation over time,

which limits the accuracy of VO [18].

Some recent research suggested scale estimation methods, but they were dedicated to pedestrians. For example, [19] [20] applied a hard constraint by having the sensors fixed to a helmet, which is not valid in some contexts. Another approach used the pedestrian face as an identified object to estimate the scale [21]. In this approach, two cameras are installed; one of them captures the user's face, while the other one is a world-facing camera. The world-facing camera is performing the VO, and the scale can be computed from the face. However, this approach requires that the pedestrian face to be static while the hand is moving. Other approaches used an average step length and a pedometer to estimate the monocular VO scale [22] [23]. Nevertheless, the step lengths and an average step length are not constant through the walk-in urban environments because the pedestrian must avoid other pedestrians, cars and wait at street crossings.

This paper introduces a novel scale recovery approach using the Spike rangefinder measurements. Our approach estimates the translation scale from the measured distances of the sequential images using Spike, which results in an accurate VO solution. Through such an ultra-low-cost sensor, our visual odometry approach can recover the scale with centimeter-level accuracy, which makes it attractive to a number of applications such as pedestrian navigation and augmented reality. This paper is structured as follows. Section 2 provides some background information about the used Spike device. Section 3 introduces the VO method used in this paper. In Section 4, the data acquisition is presented. The obtained results and some discussion are presented in Section 5. Some concluding remarks are presented in Section 6.

2. Spike Rangefinder

The Spike is a small, low-cost laser-based rangefinder device. It is typically attached to a smartphone to measure the distance to an object and then localizes it by making use of the smartphone's photo (**Figure 1**). The device pairs with the smartphone via Bluetooth, which allows it to take advantage of the smartphone's camera, GPS/GNSS, compass, and Internet connection. It comprises some features such as real-time measurements from a photo, measuring the distance between two objects, and measuring remote objects and collecting GPS/GNSS location from a distance. The first feature enables a user to obtain measurements of areas, heights, widths, and lengths instantly from a photo-taking with the device. This feature is especially useful for hard to reach objects. In the second feature, the Spike calculates the distance between two objects by aiming it at the first object and take a photo, and then pointing it at the second object and take a photo. For the third feature, the device can capture a target's coordinates (latitude, longitude, and altitude) by measuring the distance to the target and taking advantage of the smartphone's GPS/GNSS and compass.

The Spike laser rangefinder supports ranges between 2 - 200 meters, with an accuracy of ±5 cm [24]. All the measurements are saved with the image using the

Figure 1. Spike device mounted on a smartphone (https://ikegps.com/spike/).

Spike App and can be exported as a Spike file (XML format).

3. Proposed VO Approach

The VO technique used in this paper was carried out using the MATLAB computer vision toolbox. The workflow of the VO approach is presented in **Figure 2**. As a pre-processing step, the standard camera calibration is performed by taking images of a checkerboard from different positions and angles to determine the intrinsic parameters matrix of the iPhone 6 camera used in data acquisition [25] [26] [27].

In order to estimate the relative pose between sequential images, feature points are extracted through the Speeded-Up Robust Features (SURF) approach [28]. SURF is composed of three steps, namely feature extraction, feature description, and feature matching. The outliers were then excluded from the matched points, which otherwise can cause significant errors in the camera pose estimation process. The random sample consensus (RANSAC) technique is used to reject the outliers in the data [29]. The 3 × 3 fundamental matrix is then formed, which encodes the rotation and translation between two frames when an uncalibrated camera is used. The fundamental matrix is computed using the inlier point matches through the epipolar constraint, as presented in Equations (1) and (2) [26]:

$$P_i^{\mathrm{T}} F P_{(i+1)} = 0 \tag{1}$$

$$\begin{bmatrix} x_i' & y_i' & 1 \end{bmatrix} \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{pmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = 0 \tag{2}$$

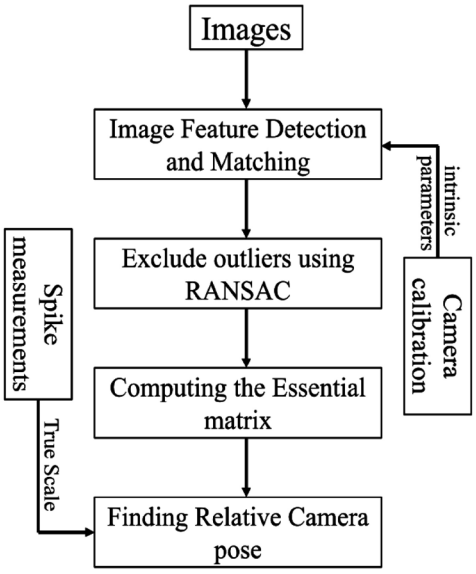


Figure 2. Detailed steps of the VO block.

where P_i and $P_{(i+1)}$ are vectors in the homogeneous coordinate system containing the detected point coordinates in the image frame (i) and its correspondence in the image frame (i+1), respectively, and (F) is the fundamental matrix. The eight-point algorithm, which requires a minimum of eight points and their correspondences, can be used to estimate the fundamental matrix since it is defined up to a scale factor. When n matched points are found, where $n > 8$, the least-squares estimation method is used to compute the fundamental matrix. In this case, Equation (3) can be re-written as:

$$\begin{bmatrix} x_1x_1' & x_1y_1' & x_1 & y_1x_1' & y_1y_1' & y_1 & x_1' & y_1' & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_nx_n' & x_ny_n' & x_n & y_nx_n' & y_ny_n' & y_n & x_n' & y_n' & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = 0 \tag{3}$$

When the intrinsic camera parameters are known, the essential matrix (E) is used, which is related to the fundamental matrix through Equation (4) [26]:

$$E = K^{\mathrm{T}}FK \tag{4}$$

where (K) is the calibration matrix of the camera system. The essential matrix has five degrees of freedom and can be decomposed using singular value decomposition to yield the relative rotation matrix (R) and the normalized translation (T) between the frames. The decomposition process is given in detail in [30].

The range measurements acquired by the Spike are used to calculate the scale factor of the relative pose. While moving in a straight line, the baseline between

two frames(S) equals the range difference between the previous frame (r_p) and the current frame (r_c), *i.e.*,

$$S = r_p - r_c \tag{5}$$

The current scaled location and orientation of the system relative to the first frame can be obtained using Equations (6) and (7). Where (C) is the current location, (P) is the previous location, (R_p) is the previous orientation, and (R_c) is the current orientation. The coordinates of the subsequent frame can then be computed relative to the previous frame. Figure 3 shows an example using three frames, where ($T1$) represents the normalized translation between the first frame and the second frame, ($S1$) is the baseline between the first and the second frames, ($S2$) is the baseline between the second frame and the third, ($R1$) the rotation between the second frame and the first frame, and ($T2$) is the coordinates of the third frame in the second frame coordinate system. Having the computed scaled trajectory, we can then construct a 3D model of the surrounding scene with the actual scale.

$$C = P + S * R_p * T \tag{6}$$

$$R_c = R * R_p \tag{7}$$

4. Data Acquisition

Two datasets were collected along precisely surveyed tracks in both of outdoor and indoor environments, as shown in Figure 4. The precise local coordinates of both tracks were estimated using observations from a Leica TS12 total station with a distance accuracy of 1 mm + 1.5 ppm, which served as the ground truth. The outdoor dataset consisted of 26 images, and the trajectory was approximately 21 meters in length. The indoor dataset, on the other hand, consisted of 61 images, and the trajectory was around 28 meters in length. The setup comprises a Spike device connected to an iPhone 6, which was mounted on a levelled pole, as shown in Figure 5.

5. Results and Discussion

The proposed approach has been tested in both of outdoor and indoor environments and processed, as explained in Section (3). The outdoor images were processed using Pix4D mapper software [31]. Two scenarios were considered in the data processing and point cloud generation. In the first scenario, the images

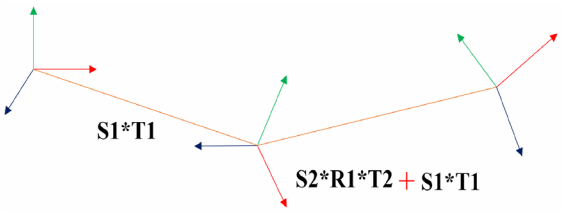
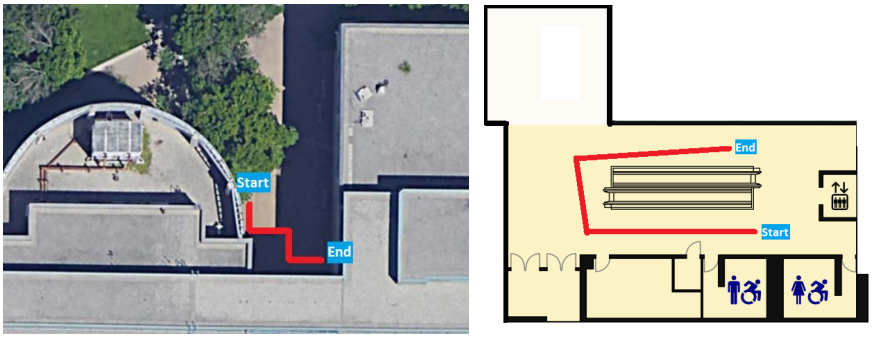


Figure 3. The orientation and position of the other frames relative to the first frame.



(a) (b)

Figure 4. Outdoor and indoor data sets layout (Ryerson University). (a) Rogers communications centre building. (b) The ground floor of the Podium building.

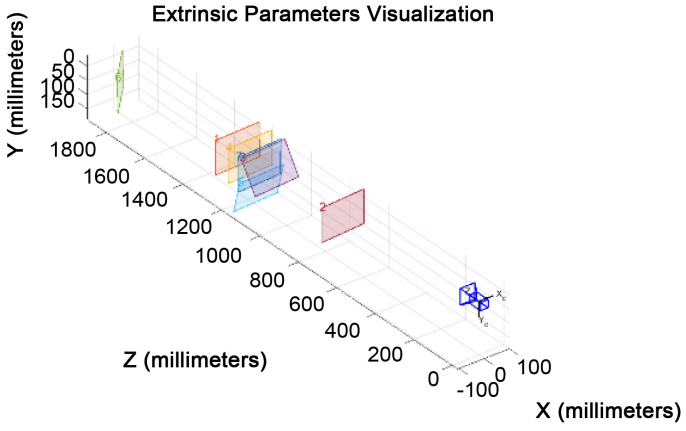
(a) (b)

Figure 5. Spike device mounted on a smartphone. (a) Spike application interface. (b) Spike device.

were geolocated using the camera poses estimated from VO after correcting for the scale factor using the Spike measurements. The generated point cloud from this scenario will be referred to as the Spike-based point cloud in the sequel. In the second scenario, the images were geolocated using iPhone GPS coordinates. The generated point cloud from this case will be referred to as the iPhone-based point cloud in the sequel. The camera calibration parameters are also estimated through the MATLAB camera calibration tool, as shown in **Figure 6**. **Figure 6(a)** shows the mean reprojection error per image, along with the overall mean error, which is found to be 0.38 pixels. The reprojection error is defined as the distance, in pixels, between the reprojected 3D points and correct image points. In general, a mean reprojection error of less than one pixel is acceptable [32]. **Figure 6(b)** provides a camera-centric view of the patterns, which examine the relative positions of the camera and the pattern to ensure that they match what is expected. As an example, a pattern that appears behind the camera indicates a calibration error.

The matching results between sequential frames before removing the outliers are shown in **Figure 7**. While the inlier matched points between the sequential

(a)



(b)

Figure 6. Calibration results using MATLAB camera calibration tool. (a) Image reprojection errors. (b) 3-D camera extrinsic parameters.

Figure 7. Matched points between sequential frames before removing the outliers.

images are shown in Figure 8.

The essential matrix can be computed using the matched points between the two frames. Then, the essential matrix is decomposed to obtain the normalized relative translation (T) and rotation matrix (R), which represents the rotation between the two frames. The following numerical example shows the mathematical steps to calculate the second frame pose relative to the first frame, assuming that the first frame coordinates are (0, 0, 0) and the first rotation matrix is the identity matrix.

The measured ranges of the first and second frames are:

$$r_p = 13.12 \text{ m}$$

$$r_c = 11.1 \text{ m}$$

Consequently, from Equation (5), the baseline can be computed as follow

$$S = r_p - r_c = 2.02 \text{ m}$$

The normalized relative translation (T) and rotation matrix (R) obtained through the decomposition of the essential matrix are:

$$T = \begin{bmatrix} -0.00307 \\ -0.05695 \\ 0.99837 \end{bmatrix}, \quad R = \begin{bmatrix} 0.99998 & 0.00417 & 0.00267 \\ -0.0041 & 0.99998 & -0.00376 \\ -0.00268 & 0.00375 & 0.99998 \end{bmatrix}$$

Using Equations (6) and (7), the current location and orientation can be obtained as:

$$\text{Current Location} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + 2.02 * \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} -0.00307 \\ -0.05695 \\ 0.99837 \end{bmatrix} = \begin{bmatrix} -0.0062 \\ -0.1150 \\ 2.016 \end{bmatrix}$$

$$\text{Current Orientation} = \begin{bmatrix} 0.99998 & 0.00417 & 0.00267 \\ -0.0041 & 0.99998 & -0.00376 \\ -0.00268 & 0.00375 & 0.99998 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0.0026 & 0.0041 \\ -0.0042 & 1 & -0.0038 \\ -0.0026 & 0.0037 & 1 \end{bmatrix}$$

By repeating the previous steps, the estimated camera poses relative to the first

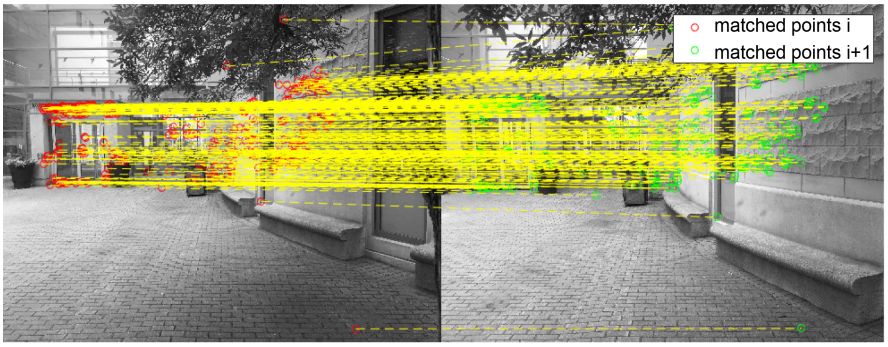


Figure 8. Point matching after outlier removal.

frame can be obtained. Figure 9 and Figure 10 show the estimated camera poses for both of the outdoor and indoor datasets, respectively. As can be seen, the camera orientations are the same, regardless of whether or not the Spike is used. However, the scale is incorrect, which is adjusted using the Spike measurements.

Figure 11 and Figure 12 compare different trajectories for both of the outdoor and the indoor datasets, respectively. In both figures, the total station-based reference trajectory is presented in red colour, while the Spike-based scaled trajectory is presented in green, and the trajectory without using the Spike is presented in blue. As can be seen, it is evident that there is an apparent drift between ground truth trajectories and the trajectories estimated from VO. This drift is likely attributed to a rotation estimation error. The root-mean-squares error (RMSE) of both datasets is presented in Table 1. The total RMSE of the outdoor data trajectory is about 70 cm, while the total RMSE of the indoor data

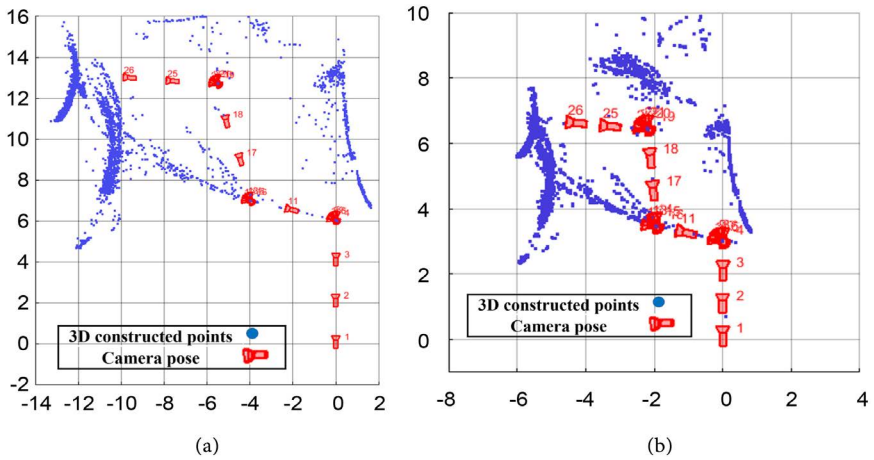


Figure 9. Camera poses for the outdoor data set. (a) Using spike measurements. (b) Without using spike measurements.

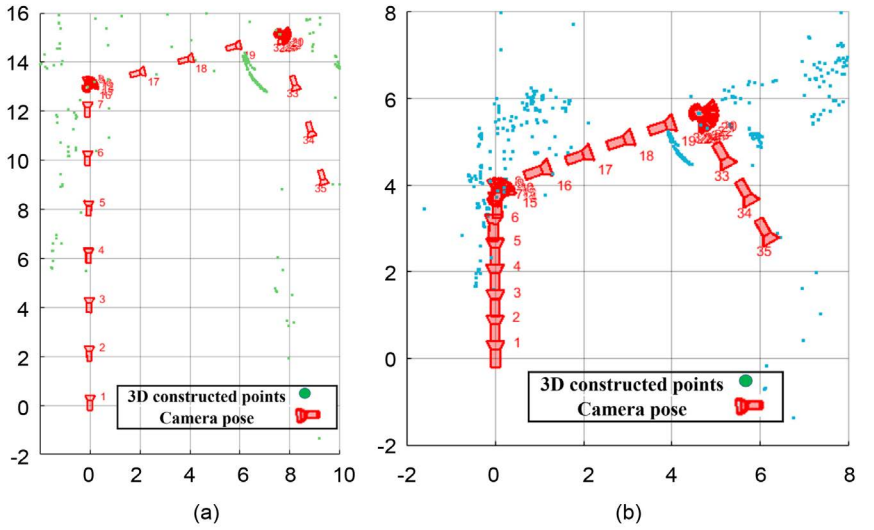


Figure 10. Camera poses for the indoor data set. (a) Using Spike measurements. (b) Without using Spike measurements.

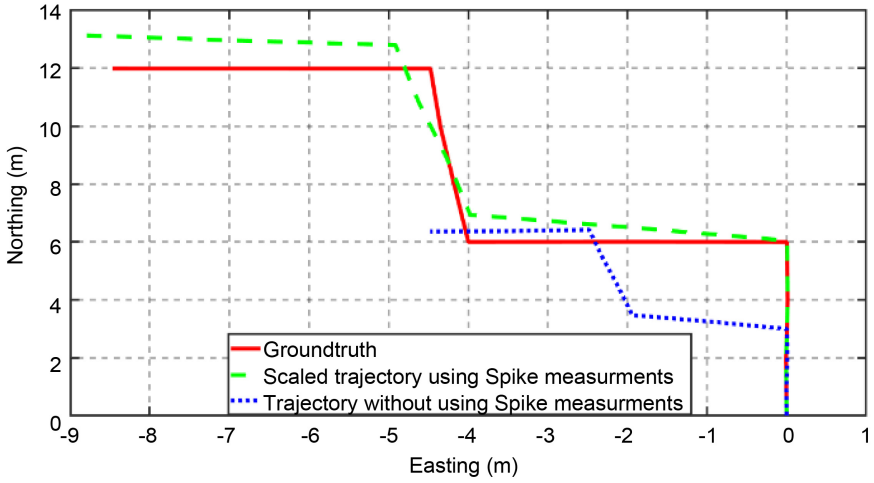


Figure 11. Comparing the different trajectories of outdoor dataset.

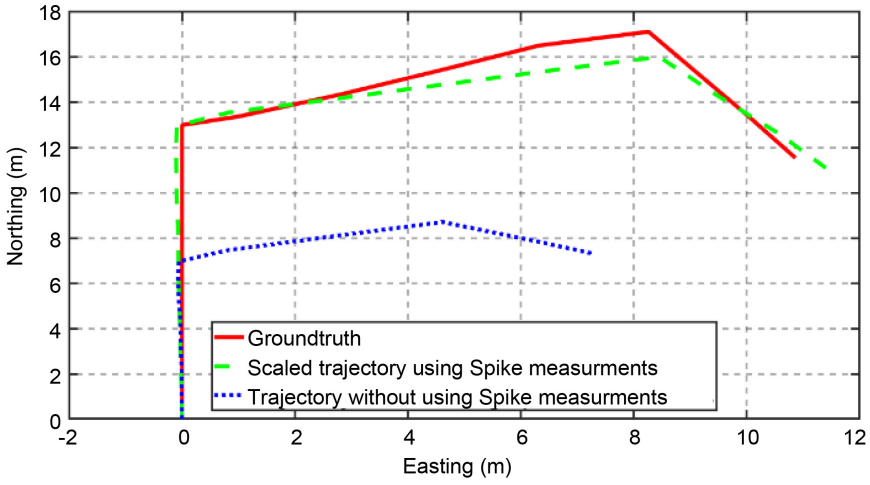


Figure 12. Comparing the different trajectories of indoor dataset.

Table 1. RMSE of both outdoor and indoor datasets.

	X_{RMSE} (meter)	Y_{RMSE} (meter)	$Total_{RMSE}$ (meter)
Outdoor dataset	0.22	0.65	0.69
Indoor dataset	0.19	0.54	0.58

is about 60 cm. However, it is observed that the estimated trajectory using Spike is close to the reference trajectory. This shows that the proposed approach using the Spike measurements allows for scale recovery of the monocular VO and precise localization of the camera. Table 2 and Table 3 compare the scale obtained from VO using the Spike with the ground truth scale measured by the total station. The average scale errors of the outdoor and indoor data sets are in the range of 1 cm and 3 cm, respectively. This shows that augmenting the range information with the monocular VO can recover the scale ambiguity to centimeter-level accuracy.

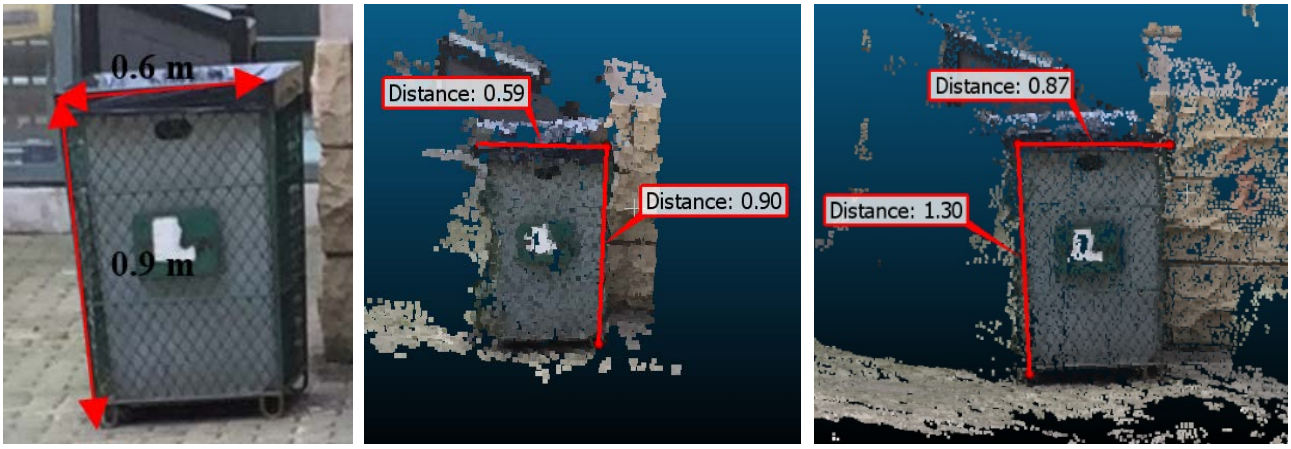
Table 2. Scale error on the outdoor dataset.

Distances between sequential Poses		Scale error
Ground Truth	VO using Spike	
2.005	2.017	0.012
2.001	2.008	0.007
1.992	2.027	0.035
2.001	2.018	0.017
1.999	2.060	0.061
2.010	1.987	−0.023
2.009	1.969	−0.041
1.989	1.985	−0.003
1.991	2.019	0.028
2.007	2.010	0.003

Table 3. Scale error on indoor dataset.

Distances between sequential Poses		Scale error
Ground Truth	VO using Spike	
2.00	2.01	0.01
2.00	1.97	−0.03
2.00	2.02	0.02
2.00	1.98	−0.02
2.00	1.99	−0.01
2.00	1.98	−0.02
1.00	1.03	0.03
1.07	1.09	0.02
2.08	1.99	−0.09
2.05	2.01	−0.04
2.05	1.98	−0.07
2.03	2.02	−0.01
2.04	1.98	−0.06
2.08	1.99	−0.09
2.04	2.01	−0.03

To further assess the effectiveness of the proposed approach, the point clouds of the two data sets were generated using the Pix4D mapper. **Figures 13-16** show the results of comparing the Spike-based and the iPhone-based point clouds. We compared the dimensions of different features from both point clouds, using CloudCompare software, with the ground truth measured in the field using a tape with 2 mm precision (**Figures 13-16**). It was found that the Spike-based point cloud is more precise than the iPhone-based counterpart. Knowing that the points were manually picked, and hence, there is also a manual measurement error in the estimated distances in **Figures 13-16**. This shows that

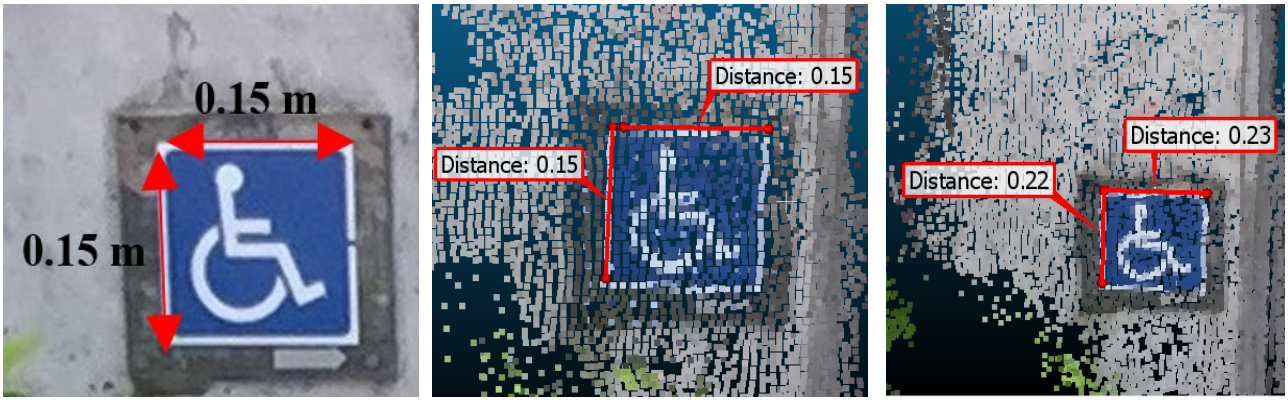


(a) (b) (c)

Figure 13. (a) Ground truth; (b) Spike point cloud; (c) iPhone point cloud.

(a) (b) (c)

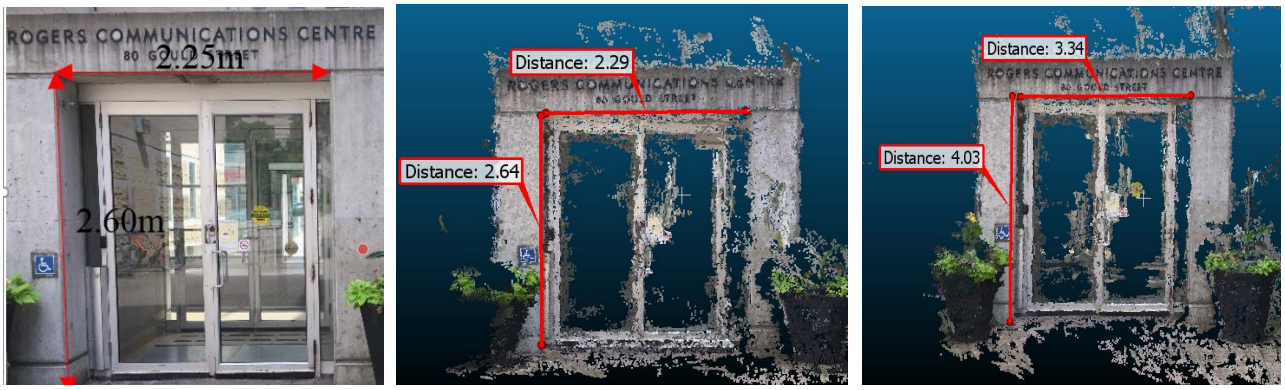
Figure 14. (a) Ground truth; (b) Spike point cloud; (c) iPhone point cloud.



(a) (b) (c)

Figure 15. (a) Ground truth; (b) Spike point cloud; (c) iPhone point cloud.

the use of Spike measurements to recover the scale ambiguity is an efficient and cost-effective approach, which significantly improves the mapping accuracy.



(a) (b) (c)

Figure 16. (a) Ground truth; (b) Spike point cloud; (c) iPhone point cloud.

6. Conclusion

In this paper, we presented a novel approach, which takes advantage of the low-cost Spike laser rangefinder to resolve the scale ambiguity in monocular visual odometry. The proposed approach was tested in both of outdoor and indoor scenarios, and the results were compared to a ground truth measured by a high-end total station. It was shown that the proposed solution allows for achieving centimeter-level accuracy in monocular VO scale recovery, which leads to an enhanced mapping accuracy.

Acknowledgements

This research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and Ryerson University. The first author would like to thank Mr. Nader Abdelaziz, a Ph.D. candidate at Ryerson University, for his help in the field data collection.

Conflicts of Interest

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

References

[1] Scaramuzza, D. and Fraundorfer, F. (2011) Visual Odometry [Tutorial]. *IEEE Robotics & Automation Magazine*, **18**, 80-92. https://doi.org/10.1109/MRA.2011.943233

[2] Nistér, D., Naroditsky, O. and Bergen. J. (2004) Visual Odometry. *Proceedings of the* 2004 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington DC, 27 June-2 July 2004, 1.

[3] Aqel, M.O., *et al.* (2016) Review of Visual Odometry: Types, Approaches, Challenges, and Applications. *SpringerPlus*, **5**, Article No. 1897. https://doi.org/10.1186/s40064-016-3573-7

[4] Mur-Artal, R. and Tardós, J.D. (2017) 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

[5] Engel, J., Schöps, T. and Cremers. D. (2014) LSD-SLAM: Large-Scale Direct Monocular SLAM. In: *European Conference on Computer Vision*, Springer, Berlin, 834-849. https://doi.org/10.1007/978-3-319-10605-2_54

[6] Davison, A.J., *et al.* (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

[7] Wang, R., Schworer, M. and Cremers, D. (2017) Stereo DSO: Large-Scale Direct Sparse Visual Odometry with Stereo Cameras. *Proceedings of the IEEE International Conference on Computer Vision*, Venice, 22-29 October 2017, 3903-3911. https://doi.org/10.1109/ICCV.2017.421

[8] Yin, X., *et al.* (2017) Scale Recovery for Monocular Visual Odometry Using Depth Estimated with Deep Convolutional Neural Fields. *Proceedings of the IEEE International Conference on Computer Vision*, Venice, 22-29 October 2017, 5870-5878. https://doi.org/10.1109/ICCV.2017.625

[9] Dang, T., Hoffmann, C. and Stiller, C. (2009) Continuous Stereo Self-Calibration by Camera Parameter Tracking. *IEEE Transactions on Image Processing*, **18**, 1536-1550. https://doi.org/10.1109/TIP.2009.2017824

[10] Klein, G. and Murray, D. (2007) Parallel Tracking and Mapping for Small AR Workspaces. 2007 6*th IEEE and ACM International Symposium on Mixed and Augmented Reality*, Nara, 13-16 November 2007, 225-234. https://doi.org/10.1109/ISMAR.2007.4538852

[11] Kitt, B.M., *et al.* (2011) Monocular Visual Odometry Using a Planar Road Model to Solve Scale Ambiguity.

[12] Choi, S., *et al.* (2011) What Does Ground Tell Us? Monocular Visual Odometry under Planar Motion Constraint. 2011 11*th IEEE International Conference on Control, Automation and Systems*, Gyeonggi-do, 26-29 October 2011, 1480-1485.

[13] Song, S., Chandraker, M. and Guest, C.C. (2015) High Accuracy Monocular SFM and Scale Correction for Autonomous Driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **38**, 730-743. https://doi.org/10.1109/TPAMI.2015.2469274

[14] Gakne, P.V. and O'Keefe, K. (2018) Tackling the Scale Factor Issue in a Monocular Visual Odometry Using a 3D City Model. *International Technical Symposium on Navigation and Timing*, Toulouse, October 2018, 228-243. https://doi.org/10.31701/itsnt2018.20

[15] Scaramuzza, D., Fraundorfer, F. and Siegwart, R. (2009) Real-Time Monocular Visual Odometry for On-Road Vehicles with 1-Point RANSAC. 2009 *IEEE International Conference on Robotics and Automation*, Kobe, 12-17 May 2009, 4293-4299. https://doi.org/10.1109/ROBOT.2009.5152255

[16] Weiss, S. and Siegwart, R. (2011) Real-Time Metric State Estimation for Modular Vision-Inertial Systems. 2011 *IEEE International Conference on Robotics and Automation*, Shanghai, 9-13 May 2011, 4531-4537. https://doi.org/10.1109/ICRA.2011.5979982

[17] Nützi, G., *et al.* (2011) Fusion of IMU and Vision for Absolute Scale Estimation in Monocular SLAM. *Journal of Intelligent & Robotic Systems*, **61**, 287-299. https://doi.org/10.1007/s10846-010-9490-z

[18] Antigny, N., *et al.* (2019) Solving Monocular Visual Odometry Scale Factor with Adaptive Step Length Estimates for Pedestrians Using Handheld Devices. *Sensors*, **19**, 953. https://doi.org/10.3390/s19040953

[19] Lupton, T. and Sukkarieh, S. (2008) Removing Scale Biases and Ambiguity from

6DoF Monocular SLAM Using Inertial. 2008 *IEEE International Conference on Robotics and Automation*, Pasadena, 19-23 May 2008, 3698-3703. https://doi.org/10.1109/ROBOT.2008.4543778

[20] Gutiérrez-Gómez, D. and Guerrero, J.J. (2013) Scaled Monocular SLAM for Walking People. *Proceedings of the* 2013 *International Symposium on Wearable Computers*, Zurich, Switzerland, September 2013, 9-12. https://doi.org/10.1145/2493988.2494351

[21] Knorr, S.B. and Kurz, D. (2016) Leveraging the User's Face for Absolute Scale Estimation in Handheld Monocular Slam. 2016 *IEEE International Symposium on Mixed and Augmented Reality*, Merida, 19-23 September 2016, 11-17. https://doi.org/10.1109/ISMAR.2016.20

[22] Leppäkoski, H., Collin, J. and Takala, J. (2013) Pedestrian Navigation Based on Inertial Sensors, Indoor Map, and WLAN Signals. *Journal of Signal Processing Systems*, **71**, 287-296. https://doi.org/10.1007/s11265-012-0711-5

[23] Tomažič, S. and Škrjanc, I. (2015) Fusion of Visual Odometry and Inertial Navigation System on a Smartphone. *Computers in Industry*, **74**, 119-134. https://doi.org/10.1016/j.compind.2015.05.003

[24] ikeGPS. Spike User Manual. https://www.gi-geoinformatik.de/wp-content/uploads/spike/Spike_User_Manual.pdf

[25] Zhang, Z. (2000) A Flexible New Technique for Camera Calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**, 1330-1334. https://doi.org/10.1109/34.888718

[26] Hartley, R. and Zisserman, A. (2003) Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511811685

[27] Mathworks. Camera Calibration and 3-D Vision. https://www.mathworks.com/help/vision/camera-calibration-and-3-d-vision.html

[28] Bay, H., Tuytelaars, T. and Van Gool, L. (2006) SURF: Speeded Up Robust Features. In: *European Conference on Computer Vision*, Springer, Berlin, 404-417. https://doi.org/10.1007/11744023_32

[29] Torr, P.H. and Zisserman, A. (2000) MLESAC: A New Robust Estimator with Application to Estimating Image Geometry. *Computer Vision and Image Understanding*, **78**, 138-156. https://doi.org/10.1006/cviu.1999.0832

[30] Georgiev, G.H. and Radulov, V.D. (2014) A Practical Method for Decomposition of the Essential Matrix. *Applied Mathematical Sciences*, **8**, 8755-8770. https://doi.org/10.12988/ams.2014.410877

[31] Pix4D. https://www.pix4d.com

[32] Scaramuzza, D., Martinelli, A. and Siegwart, R. (2006) A Toolbox for Easily Calibrating Omnidirectional Cameras. 2006 *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, 9-15 October 2006, 5695-5701. https://doi.org/10.1109/IROS.2006.282372