DEVELOPMENT OF A SINGLE-VIEW ODOMETER BASED ON PHOTOGRAMMETRIC BUNDLE ADJUSTMENT

S. J. Yoon [1], W.S. Yoon [1], J.W. Jung [2], T. Kim [1,*]
[1] Dept. of Geoinformatic Engineering, Inha University, 100 Inharo, Namgu, Incheon, Republic of Korea
- ymh5648@naver.com, rainydayz@inha.edu, tezid@inha.ac.kr
[2] KT Institute of Convergence Technology, 151 Taebong-ro, Seocho-gu, Seoul, Republic of Korea
- jongwon.jeong@kt.com

Commission II, WG II/5

KEY WORDS: Single-view odometer, Photogrammetry, Bundle Adjustment, Scale Calibration, KITTI Community

ABSTRACT:

Recently, a vehicle is equipped with various sensors, which aim smart and autonomous functions. Single-view odometer estimates its pose using a monoscopic camera mounted on a vehicle. It was generally studied in the field of computer vision. On the other hands, photogrammetry focuses to produce precise three-dimensional position information using bundle adjustment methods. Therefore, this paper proposes to apply photogrammetric approach to single view odometer. Firstly, it performs real-time corresponding point extraction. Next, it estimates the pose using relative orientation based on coplanarity conditions. Then, scale calibration is performed to convert the estimated translation in the model space to the translation in the real space. Finally, absolute orientation is performed using more than three images. In this step, we also extract the appropriate model points through verification procedure. For experiments, we used the data provided by KITTI (Karlsruhe Institute of Technology and Toyota Technological Institute) community. This technique took 0.12 seconds of processing time per frame. The rotation estimation error was about 0.005 degree per meter and the translation estimation error was about 6.8%. The results of this study have shown the applicability of photogrammetry to visual odometry technology.

1. INTRODUCTION

Visual odometry estimates the pose of the platform where the camera is mounted using only the image. This technology has been used in a variety of fields such as smart cars, VR (Virtual Reality), and robotics. Single-view odometer estimates its pose using a monoscopic camera. This is a very efficient technique because it requires only one camera. Currently, single view odometer studies are mainly based on the computer vision technique. Among them, FTMVO (Fast Techniques for Monocular Visual Odometry) technology estimates the pose using the 5-point based essential matrix developed in computer vision(Mirabdollah and Mertsching, 2015). It estimates the geometry of images by calculating essential matrix and decomposing it into the position and attitude of camera. On the other hands, photogrammetric technology estimates directly the camera’s position and attitude by using bundle adjustment methods, such as relative orientation and absolute orientation. For photogrammetry, the focus is to produce precise three-dimensional position information. Therefore, we propose to apply photogrammetric bundle adjustment to single view odometer technology. This paper is structured as follows. Section 2 describes the proposed method. Experiments and results are explained in Section 3. Finally, Section 4 deals with the conclusions of this study.

2. PROPOSED METHOD

The purpose of this study is to develop real-time single-view odometer based on photogrammetric method. First, corresponding point extraction is performed. In this step, a lot of processing time is consumed and the number of corresponding points affects the accuracy. Therefore, we compare the number of extracted points per processing time of several methods and choose the most appropriate one. Next, relative orientation is performed to estimate pose. To fit MMS (Mobile Mapping System) geometry, we derived the observation equation by setting the optical axis as the baseline. Next scale calibration is performed since the results are on the model space. Then, in order to verify the model points, we project them onto the image and compare the projected and the actual image points. Based on the comparison result, it is judged which model point is suitable. The last step is absolute orientation. This step is performed when three or more images have been provided. The estimated pose is in the real space after absolute orientation. Finally, the trajectory is determined by accumulating the estimated poses.

2.1 Corresponding point extraction

There are two methods of corresponding point extraction in feature based matching: feature matching method and feature tracking method. Feature matching method extracts the feature points from the two images and directly compares the resemblance between these feature points. This proceeds in the order of detection, description and matching. Feature tracking method extracts feature points from one image and tracks the corresponding points from other images. This method is faster than the feature point matching method because it does not need to perform additional descriptor calculation or similarity calculation between descriptors. However, it has the limitation that the displacement of feature points between images should be small. In order to select the best extraction method, we checked corresponding points per unit time of several methods. These methods are provided by OpenCV and the combinations are as in Table 1.

[*] Corresponding author

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLII-2-1219-2018 | © Authors 2018. CC BY 4.0 License.

Detector	Descriptor	Matcher
SIFT	SIFT	BruteForce
SURF	SURF	BruteForce
FAST	BRISK	FLANN
FAST	ORB	FLANN
FAST	FREAK	FLANN

Extractor	Tracker
FAST	Kanade-Lucas-Tomasi tracker
Shi-Thomasi corner	Kanade-Lucas-Tomasi tracker

Table 1. The corresponding points extraction methods

2.2 Relative orientation and scale calibration

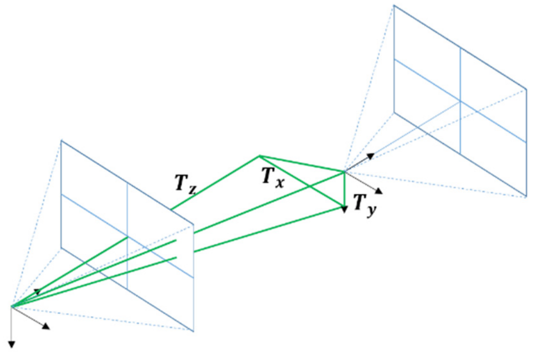


Figure 2. MMS geometry

The relative geometry of two images is estimated based on coplanarity conditions in photogrammetry(Kim and Kim, 2016). This is called relative orientation. It sets one of the translations representing the geometry of the two images as the baseline. This value determines the scale of the model space. Then, it estimates the relative translation and rotation, which are usually called EOP (Exterior Orientation Parameters). As shown in Equation (1), the optical axis was set as a baseline because the translation of optical axis was most prominent in MMS geometry(Jeong and Kim, 2017) as Figure 2.

$$\begin{bmatrix} \frac{\delta F}{\delta T_x} & \frac{\delta F}{\delta T_y} & \frac{\delta F}{\delta \omega} & \frac{\delta F}{\delta p} & \frac{\delta F}{\delta k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \Delta T_x \\ \Delta T_x \\ \Delta \omega \\ \Delta p \\ \Delta k \end{bmatrix} = \begin{bmatrix} -F_0 \\ \vdots \end{bmatrix} \quad (1)$$

where F = coplanarity equation
T_x, T_y, T_z = translation
ω, p, k = rotation
F_0 = observation value

Since the baseline can be set to any value, the estimated pose is not an actual physical quantity. Therefore, it is necessary to correct the scale at the baseline to calculate the actual translation. The scale calibration method assumes that the height from the ground to the camera is known in advance. First, we extracted points from the bottom of the image and randomly selected three points, which were generated based on previously estimated geometry information. Then, we constructed the ground surface on the model space and calculated the height to the camera.

In the model space, the ground surface was determined by establishing a plane equation using model points as Equation (2).

$$\vec{a} = \overrightarrow{p_1 p_2} = \langle a_1, a_2, a_3 \rangle,\ \vec{b} = \overrightarrow{p_2 p_3} = \langle b_1, b_2, b_3 \rangle$$

$$\vec{a} \times \vec{b} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix} = \begin{vmatrix} a_2 & a_3 \\ b_2 & b_3 \end{vmatrix} \vec{i} - \begin{vmatrix} a_1 & a_3 \\ b_1 & b_3 \end{vmatrix} \vec{j} + \begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix} \vec{k} \quad (2)$$

$$\rightarrow A\vec{i} + B\vec{j} + C\vec{k}$$

$$Ax + By + Cz + D = 0$$

where p_1 , p_2 , p_3 = model points
D = constant

Next, assuming the above assumption, we estimated the actual scale by comparing the calculated height with actual height. Scale estimation is shown in Equation (3). By applying the estimated scale to translation of the EOP, we converted its size to the actual physical size.

$$1 : S = H_{model} : H_{true}$$
$$S = \frac{H_{true}}{H_{model}} \quad (3)$$

where S = scale for correction
H_{model} = height to camera on model space
H_{true} = height to camera on actual space

2.3 Corresponding point verification

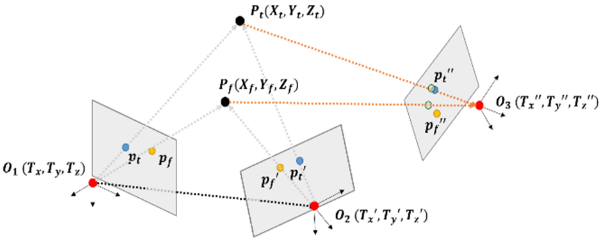


Figure 3. Verification of corresponding point

MMS images contain moving objects such as human, and vehicles. Often they are extracted as features and not eliminated as outliers. They reduce the accuracy of estimation. Figure 3 explains corresponding point verification. When the true model point (P_t) is projected onto O_3, it has a small separation from the image point (p_t''). However, in the case of the false model point (P_f), it shows a large separation when projected onto O_3. Therefore, we projected the model points onto the images and calculated the separation distance from the corresponding image points. Based on the results, the appropriate model point was extracted.

https://doi.org/10.5194/isprs-archives-XLII-2-1219-2018 | © Authors 2018. CC BY 4.0 License.

2.4 Pose estimation by absolute orientation

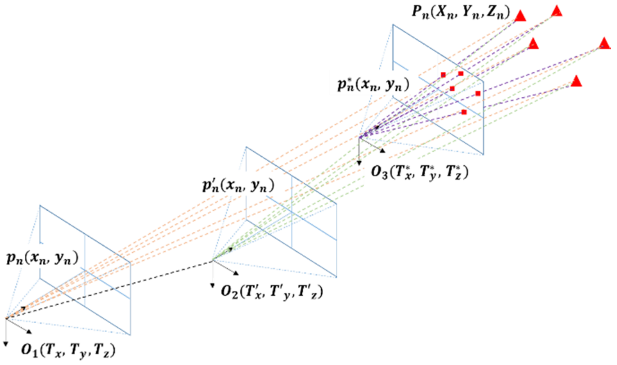


Figure 4. Absolute orientation in MMS geometry

If more than three images can be used, absolute orientation can be applied to estimate poses. Figure 4 represents the geometry of the image moving from the bottom left to the top right. First, we extracted p_n^* corresponding to image points p_n and p_n'. Then we determined model points (P_n) using previously estimated EOPs between O_1 and O_2. The absolute orientation was based on the collinearity equation, which is the condition that each projection center of two images and one object are on the same plane. Therefore, it was established through the relationship of model points (P_n) and image points (p_n^*) as in Equation (4).

$$F = x_n^* - f\left[\frac{r_{11}(X_{P_n} - T_x^*) + r_{12}(Y_{P_n} - T_y^*) + r_{13}(Z_{P_n} - T_z^*)}{r_{31}(X_{P_n} - T_x^*) + r_{32}(Y_{P_n} - T_y^*) + r_{33}(Z_{P_n} - T_z^*)}\right] \quad (4)$$

$$G = y_n^* - f\left[\frac{r_{21}(X_{P_n} - T_x^*) + r_{22}(Y_{P_n} - T_y^*) + r_{23}(Z_{P_n} - T_z^*)}{r_{31}(X_{P_n} - T_x^*) + r_{32}(Y_{P_n} - T_y^*) + r_{33}(Z_{P_n} - T_z^*)}\right]$$

where f = focal length
X_n, Y_n, Z_n = model point (P_n) coordinates
x^*_n, y^*_n = image point (p_n^*) coordinates
T^*_x, T^*_y, T^*_z = Translation
$r_{11\sim33}$ = 3×3 rotation matrix element
n = 1 to the number of corresponding point

Then, each geometric element was estimated through an iterative least square method and defined in the space of the model points (P_n).

3. EXPERIMENTS AND ANALYSIS

3.1 Experiment data

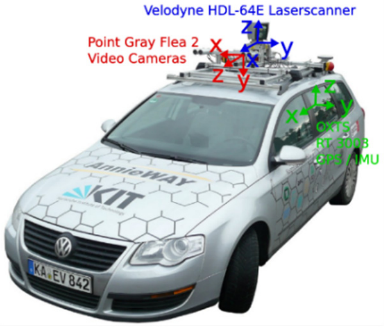


Figure 5. Fully equipped platform of KITTI

Figure 6. KITTI image example

The experiment used the open dataset provided by the KITTI community(Geiger et al., 2013). This includes the images, the true values of the poses, and the actual camera height acquired by the vehicle as shown in Figure 5. The images were taken with an optical lens with a viewing angle of about 90 degrees as Figure 6 and a Sony ICX267 with 1.4 megapixels. Nine sequences were utilized in the experiments.

3.2 Results and analysis

First, the corresponding point extraction experiment was performed. In the experiment, we used the first 10 images in each sequence and calculated the averages. The results are summarized in Table 7 and 8. In Table 8, the methods for (a) through (g) are the same as those in Table 7.

Method		Number of points
(a)	SIFT-SIFT-BruteForce	851.6
(b)	SURF-SURF-BruteForce	1013.9
(c)	FAST-BRISK-FLANN	1886.7
(d)	FAST-ORB-FLANN	2298.2
(e)	FAST-FREAK-FLANN	2298.2
(f)	FAST-KLT tracker	2486.8
(g)	Shi-Thomasi corner-KLT tracker	1739.7

Table 7. Corresponding point extraction results

Method	Processing time	Number of points per unit time (num. /sec.)
(a)	0.69	1223.30
(b)	0.40	2635.92
(c)	0.13	15209.61
(d)	0.08	29101.80
(e)	0.08	28143.88
(f)	0.03	94782.52
(g)	0.01	180708.18

Table 8. Corresponding point extraction processing times

When FAST detection was used, a relatively large number of corresponding points were extracted, such as (c) to (f) in Table 7. However, in the case of feature matching, the processing time was longer as in Table 8. Fast detection extracted more points than the Shi-Thomasi corner method, but the processing time was increased by three times as (f) and (g). Finally, we confirmed the fastest processing times when extracting feature points using the Shi-Thomasi corner method and tracking by KLT tracker and applied it to our proposed method.

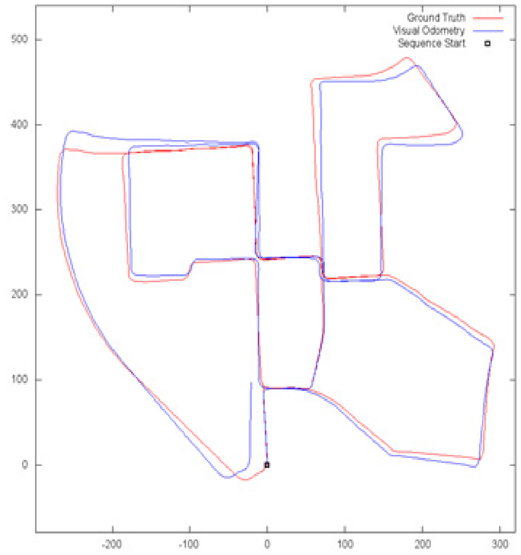


Figure 9. Estimated trajectory by relative orientation

The next is the result of applying relative orientation and scale calibration to visual odometry. Figure 9 shows the experimental results for sequence 00 of KITTI data. The red line represents the true value, and the blue line is the estimation result. As a result of the experiment in 9 sequences, an average rotation error of 0.05 degree per meter occurred. In the case of translation, the error rate was 5.46%.

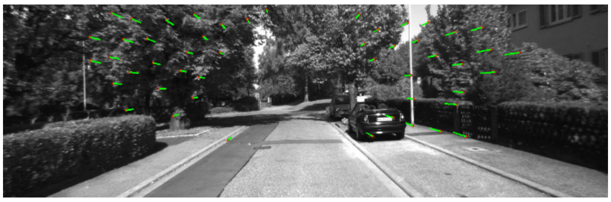


Figure 10. Sample image of ideal feature movement

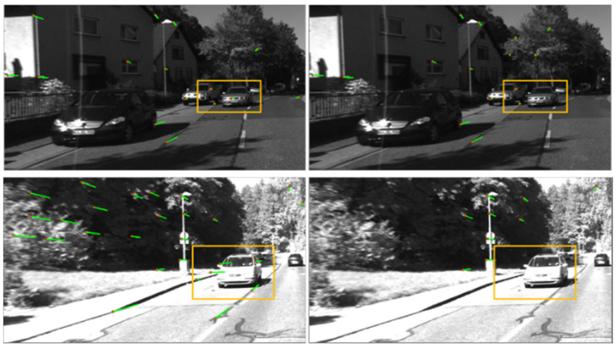


Figure 11. Comparison before and after corresponding point verification (Left (a): before, Right (b): after)

In Figure 10 and 11, the green line indicates the feature movement direction between previous and current images. The red dot indicates the head direction. Ideally, the feature motion vector should appear to spread out around one point as Figure 10. However, as mentioned in Section 2.3, abnormal feature motion vectors were generated when the vehicle passed as (a) in Figure 11. We confirmed that although the number of points was reduced, unstable points were eliminated as (b) in Figure 11.

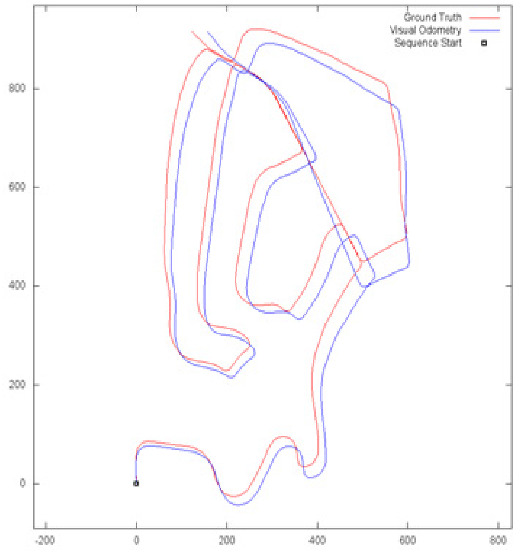
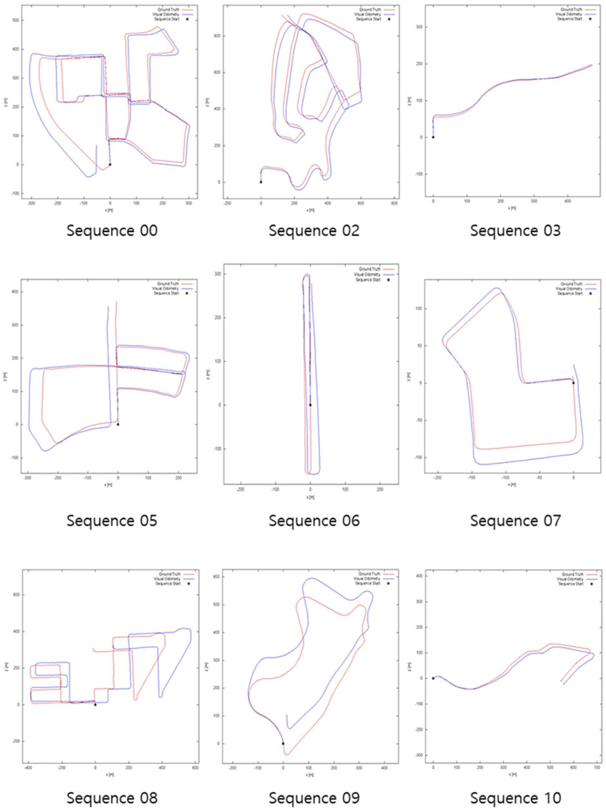


Figure 12. Estimated trajectory by proposed method

Figure 12 shows the result of the proposed method in sequence 02. We used five images to verify the corresponding points and estimate the pose. As a result of the experiment in 9 sequences, an average rotation error was 0.005 degree per meter and the translation error rate was 6.8%. It took about 0.12 seconds processing time per frame.



Sequence 00
Sequence 02
Sequence 03
Sequence 05
Sequence 06
Sequence 07
Sequence 08
Sequence 09
Sequence 10

Figure 13. All experimental results by proposed method

In case of sequence 08, a large error appeared to have occurred as shown in Figure 13. However, it was the result that the error which occurred at the beginning was accumulated continuously. Numerically, the translation error was 5.43% and the rotation error was 0.005 degree per meter in sequence 08.

4. CONCLUSION

In this study, we proposed a real-time single-view odometer based on photogrammetric techniques. The study consisted of five steps: corresponding point extraction, relative pose estimation by relative orientation, scale calibration, absolute orientation, and corresponding point verification. The developed odometer has a processing speed of about 0.12 seconds per frame, which was suitable for real time processing. In all the sequences tested, the rotation error was very small. The translation errors were also small, except the case where corresponding points were sparse. Additional study is required in the corresponding point extraction step. In particular, it was difficult to extract feature points on images of highway. In this case, the number of initial features was small and a large portion of outliers was included. In order to improve the accuracy, we will carry out further studies to extract proper feature points. The results of this study have shown the applicability of photogrammetry to visual odometry technology.

ACKNOWLEDGEMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. NRF-2016R1A2B4013017).

REFERENCES

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R., 2013. Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, 32(11), pp. 1231-1237.

Jeong, J. and Kim, T., 2017. Comparison of Essential Matrix and Relative Orientation Based Vehicle Motion Estimation in Monocular Visual Odometry. In: *Proceedings of the KSRS Fall Conference*, Yesan, Republic of Korea, pp. 422-425.

Kim, J., Kim, T., 2016. Comparison of computer vision and photogrammetric approaches for epipolar resampling of image sequence. *Sensors*, 16(3), 412.

Mirabdollah, M. H., and Mertsching, B., 2015. Fast techniques for monocular visual odometry. In: *German Conference on Pattern Recognition*, Springer, Cham, pp. 297-307.