Modeling of Unbounded Long-Range Drift in Visual Odometry

Ruyi Jiang
Shanghai Jiao Tong University
Shanghai, China
jiangruyi@sjtu.edu.cn

Reinhard Klette
University of Auckland
Auckland, New Zealand
r.klette@auckland.ac.nz

Shigang Wang
Shanghai Jiao Tong University
Shanghai, China
wangshigang@sjtu.edu.cn

Abstract

Visual odometry is a new navigation technology using video data. For long-range navigation, an intrinsic problem of visual odometry is the appearance of drift. The drift is caused by error accumulation, as visual odometry is based on relative measurements, and will grow unboundedly with time. The paper first reviews algorithms which adopt various methods to suppress this drift. However, as far as we know, no work has been done to statistically model and analyze the intrinsic properties of this drift. This paper uses an unbounded system model to represent the drift behavior of visual odometry. The model is composed of an unbounded deterministic part with unknown constant parameters, and a first-order Gauss-Markov process. A simple scheme is given to identify the unknown parameters as well as the statistics of the stochastic part from experimental data. Experiments and discussions are also provided.

1. Introduction

Visual odometry uses camera(s) to incrementally calculate a robot's motion between frames, and finally position the robot in the 3D world. In general, it can determine the ego-motion in all six degrees of freedom in the 3D world. Though there are some other sensors available for navigation, such as odometry, GPS, IMU and so forth, visual odometry has its own advantages. Classical odometry, which is installed on the robot's wheel axis, is usually deceived by wheel slippage, especially in outdoor environments. Also, it is generally incapable of 6D motion estimation. GPS is not always available for navigation, due to signals being missing or jammed. A typical example is the successful application of visual odometry for NASA's MER missions [2]; in that case GPS is completely impossible. Also, compared to GPS and IMU, cameras as used in visual odometry are relatively cheap.

With features as stated above, visual odometry has already been widely tested or applied in many fields. For driver assistance or autonomous driving, the ego-motion of the vehicle can be obtained by analyzing the video input from the camera(s) installed in the vehicle [4, 7]. Visual odometry is also a popular choice in Simultaneous Localization and Mapping (SLAM) to obtain the motion of a robot. By looking downward, visual odometry assists a helicopter to calculate its own moving trajectory [5]. It is even applied in underwater situations to help a robot to navigate [9].

Various algorithms have been tested to implement visual odometry using monocular [10, 12] or stereo [4, 5, 11], or perspective or omnidirectional [10, 12] cameras. The most popular framework for visual odometry is based on feature matching and tracking, which adopts a general work flow as shown in Fig. 1. While considering that feature-based methods are sensitive to systematic errors due to intrinsic and extrinsic camera parameters, appearance-based visual odometry uses appearance of the world to extract motion information (e.g., [10]). Recently, a direct method was also tested for visual odometry with very accurate results [3]. Furthermore, many other sensors can also be integrated with cameras to provide a more accurate result for navigation (as in [8]).

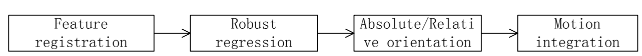

Figure 1. General work flow of feature-based visual odometry.

Though visual odometry can achieve very accurate results in short distance, one intrinsic problem of visual odometry among all the algorithms is its drift in long-range navigation, without help of global sensors like GPS. The drift is caused by error accumulation, as visual odometry is based on the relative measurement, and will increase unboundedly with time. Relative motion matrices between frames are concatenated to produce the final position, during which small errors in these matrices accumulate to a large amount, and the distance measurement will drift from its real trajectory after some long time navigation. For feature-based algorithms, the sources for these small errors are mainly uncertainties of feature localization and triangulation.

Section 2 will give a brief review of the various methods adopted in visual odometry algorithms to suppress the drift. However, as far as we know, no work has been reported about statistical modeling and analyzing the intrinsic properties of this drift. Section 3 represents the drift by an unbounded system model combining a deterministic part and a first-order Gauss-Markov process. Identification of the model parameters, using a simple strategy, is also introduced. Experiments and discussions are provided in Sections 4 and 5.

2. State of the art

Before proceeding to drift-minimization algorithms, we discuss at first a method to quantify drift. Currently, the *offset ratio* (OR), ratio of the final drift value to the traveled distance, is the common choice to measure the drift when running a visual odometry algorithm over some distance, from tens or hundreds of meters to several kilometers. (Drawbacks of OR, and a better quantification method will be discussed later in this paper.)

Note that the following review of algorithms is not for visual odometry, but for drift-minimization methods along with motion catenation adopted in these algorithms. It has been proved and widely accepted that integrating visual odometry with other positioning sensors, such as gyro or GPS, will reduce the drift to a great extent. But this is not the problem to be discussed in this paper. As visual odometry alone has its practical and theoretical meanings, the paper analyzes drift without any help from additional sensors. Obviously, increasing the accuracy of the estimated motion vectors in every time step will definitely slow down the growth of the overall drift. This kind of drift-minimization is also not discussed in the paper.

A "fire wall" was inserted into sequences in [7] to protect against error propagation. With fire walls, those relative poses estimated before the fire wall will only affect the choice of the coordinate system for subsequent poses, and the relative poses after a fire wall will be estimated as if the system is started afresh. It is supposed that fire walls suppress the propagation of gross errors and slow down the error buildup. From the provided experimental results, visual odometry had an accuracy, compared to the ground truth measured by a Differential Global Positioning System (DGPS), of 1.07%, 4.86%, and 1.63%, for three outdoor runs with a traveled distance of 185.88, 266.16, or 365.96 meters, respectively.

Bundle adjustment is another scheme that can be adopted to suppress error accumulation. It is widely used for off-line structure and motion or SLAM problems. Full bundle adjustment is almost impossible for on-line long range navigation, as there will be a huge number of poses and features to be optimized. A sliding-window sparse bundle adjustment was applied in [11] for visual odometry. A subset of several images (the number is fixed, or adaptive to the motion vector) is continuously selected to perform bundle adjustment. Experiments on some simulated data show that sparse bundle adjustment slows down the drift.

Though visual odometry uses commonly various methods to suppress the drift, no work has been done to clearly model it. The authors of [8] analyzed the contribution of position and orientation errors to the overall drift, and observed that the drift will not grow linearly in the distance traveled, but super-linearly. The growth was regarded as $O(dist^{\frac{3}{2}})$, but no specific models and parameters were provided. In order to eliminate this super-linear drift, an absolute orientation sensor was used to provide periodic updates to the orientation estimate. Simulations indicated that less than 1% OR is achievable if an absolute orientation sensor is integrated.

As a new and promising sensor, visual odometry needs a methodology for systematic and comparative analysis of its drift, in order to quantify the performance of various algorithms. For this purpose, OR has its drawbacks. First, drift will not increase linearly with the distance traveled, which was stated in [8] and will be further proved in this paper. Thus, OR from running algorithms on some distances will change with the different distances traveled. Moreover, running the same algorithms on the same dataset repeatedly will produce quite different ORs. The reason is that drift is a random process, and it will not always increase, but sometimes it decreases at some places, as errors in different motion vectors will compensate each other to some extent. Thus, using end-point values (the final drift values, and the final traveled distances) is incapable to model the whole random process. An example is shown in Fig. 2. Consid-

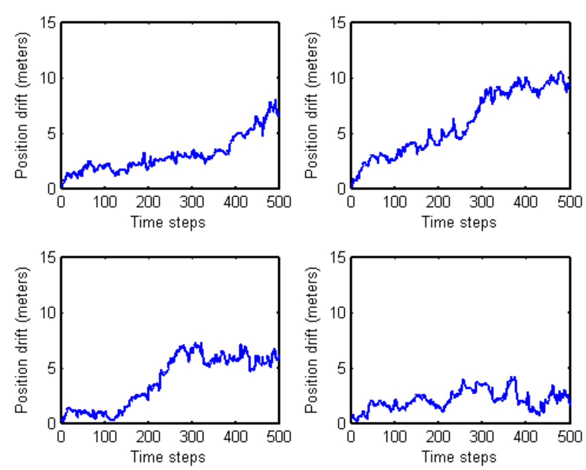

Figure 2. Position drifts after running the same visual odometry with simulated data for the same time steps. Note that drift values can be quite different, which results into incapability of the offset ratio.

ering these findings, a more accurate quantification method will be introduced in this paper.

3. Drift model and identification

In this paper, coordinate frame transformations are used to represent both poses and motions. Using a general notation, a pose E is the transformation from the world coordinate frame to that of the camera, and a motion M is the transformation of the coordinate frame of the camera between time t to $t+1$. As drift in orientation is limited to a range of $[-\pi, \pi]$, and will finally contribute to a drift in position, thus only positional drift is considered in this paper.

The concatenated camera pose at time t is denoted by E_t, and the estimated motion from time t to $t+1$ is denoted by M_t. Then,

$$E_{t+1} = E_t \cdot M_t \tag{1}$$

Note that the multiplication of M_t from the right is because the motion M_t is relative to the camera coordinate frame at t. A general algebraic structure of E and M is

$$\begin{bmatrix} R_{3\times3} & T_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}_{4\times4} \tag{2}$$

where $R_{3\times3}$ is the rotational matrix, and $T_{3\times1} = [x, y, z]^T$ is the translational vector. Considering the translational drift, let

$$\Delta d_{t+1} = \|d_{t+1} - \bar{d}_{t+1}\| \tag{3}$$

where $\|\cdot\|$ is the Euclidean distance, and d_{t+1} and $\bar{d}_{t+1}$ are the camera's estimated and true positions at time $t+1$, respectively.

3.1. Drift model

The drift in Eq. 3 is a nonstationary random process, and will increase unboundedly with time. We adopt a similar model as introduced in [6] to describe the long-range unbounded drift d in visual odometry. This model was considered as a general tool for a wide range of navigation instruments, but it was never introduced to visual odometry. For visual odometry, the drift model is

$$\ln \Delta d_t = F_t \cdot \mathbf{p} + u_t \tag{4}$$

where $F_t = [1, \ln d_t]$ is an unbounded deterministic matrix, $\mathbf{p} = [a, b]^T$ is a constant parameter vector to be identified, and u_t is a zero mean stochastic process. As noted in [6], u_t can be a correlated nonstationary process. The model is composed of two parts: a deterministic component $F_t \cdot \mathbf{p}$ and a stochastic component u_t.

Without considering the stochastic process u_t, we have the following from Eq. 4:

$$\Delta d_t = e^a \cdot d_t^b \tag{5}$$

This equation describes the relationship between the drift and the traveled distance (or camera position). The rationality of this model lays in the fact that the drift will increase exponentially with the distance traveled, and will be unbounded, as revealed in the current research.

3.2. Parameter identification

The identification of this drift model is then given by estimating $\mathbf{p}$ and characterizing u_t from Δd_t. The steps of the whole identification process are as follows:

1. Estimate $\mathbf{p} = [a, b]^T$ assuming that u_t equals zero. The estimated parameter vector is denoted by $\bar{\mathbf{p}}$.
2. Estimate the sample function of the stochastic process u_t using
$$\bar{u}_t = \Delta d_t - F(t) \cdot \bar{\mathbf{p}} \tag{6}$$
3. Characterize the process u_t using $\bar{u}_t$.

Before describing those steps further in detail, we assume that $\hat{\mathbf{p}}$ is the true parameter vector for the measurement Δd_t. The available number of measurements equals N, which is assumed to be relatively large. As we are modeling an unbounded drift, a larger value of N means that parameters are estimated more accurately.

3.2.1 Estimation of $\bar{\mathbf{p}}$

If u_t is equal to zero then the parameter estimate $\bar{\mathbf{p}}$ is defined to be that $\mathbf{p}$-value which minimizes

$$J(\mathbf{p}) = \sum_{t=1}^{N} (\Delta d_t - F_t \mathbf{p})^T (\Delta d_t - F_t \mathbf{p}) \tag{7}$$

Using a least-squares approach, the solution is defined by

$$\bar{\mathbf{p}}(N) = M^{-1}(N) \cdot \sum_{t=1}^{N} F_t^T \Delta d_t \tag{8}$$

where

$$M(N) = \sum_{t=1}^{N} F_t^T F_t \tag{9}$$

As F_t is an unbounded matrix, [6] proved that the smallest eigenvalue of $M(N)$ will grow at least as fast as N when N goes towards infinity. The same paper concluded that $M^{-1}(N)$ exists, and its largest eigenvalue will decrease towards zero at least as fast as N^{-1} if N goes towards infinity.

As N measurements are used to model the whole unbounded system, a convergence problem must be considered. The covariance $P(N)$ of $\bar{\mathbf{p}}$ is equal to

$$\begin{aligned} P(N) &= E(\hat{\mathbf{p}} - \bar{\mathbf{p}}(N))(\hat{\mathbf{p}} - \bar{\mathbf{p}}(N))^T \\ &= M^{-1}(N) \left\{ \sum_{t,s=1}^{N} F_t^T E[u_t u_t^T] F_s \right\} M^{-1}(N) \end{aligned} \tag{10}$$

Dataset	Method	e^a	b	τ	σ_u^2
Simulated	ABSOLUTE	0.025	1.20	110	11.7
	SBA	0.0025	0.98	44	8e-8
Real	ABSOLUTE	0.0103	1.23	232	1.34
	SBA	0.000342	1.64	290	0.60

Table 1. Drift model parameters as identified in our experiments.

[6] proved almost sure (AS) convergence and mean square (MS) convergence of the estimated $\bar{\mathbf{p}}(N)$ towards $\hat{\mathbf{p}}$ as N approaches infinity.

3.2.2 Characterization of $\bar{u}_t$

When the parameter vector $\bar{\mathbf{p}}$ is estimated by Eq. 8, the sampled process can be calculated from Eq. 6. Then a first-order Gauss-Markov process can be used to characterize the sampled process $\bar{u}_t$ as follows:

$$\bar{u}_t = (1 - \frac{1}{\tau})\bar{u}_{i-1} + \omega_n \tag{11}$$

where τ is a constant called *correlation time*, and ω_n is the *driving noise* modeled as zero-mean wide-band noise with variance σ_n^2. The variance of the Gauss-Markov process σ_u^2 equals

$$\sigma_u^2 = \sigma_n^2/(\frac{2}{\tau} + \frac{1}{\tau^2}) \tag{12}$$

The parameters for the $\bar{u}_t$-process are given by τ and σ_u^2, and both can be identified from experimental data by stochastic methods. Parameter σ_u^2 is the variance of the sampled functions $\bar{u}_t$. The time constant τ can be estimated from $\bar{u}_t$'s autocorrelation data. This is because the first-order Markov process has an autocorrelation known as

$$R_u(t) = \sigma_u^2 e^{-t/\tau} \tag{13}$$

For the normalized autocorrelation $\bar{R}_u(t)$ (normalization means $\bar{R}_u(0) = 1$), we have that $\tau = t$ when $\bar{R}_u(t) = e^{-1}$. In this way, the time constant τ can be estimated which is the value of t corresponding to the normalized autocorrelation value 0.368 (i.e., e^{-1}). For an example of parameter identification, see the experimental section below.

3.3. Drift quantification

There are four parameters in the drift model as established: a, b, τ, and σ_u^2. Among those, a and b can be used to quantify the drift for various visual odometry algorithms. The value of a (then used in e^a) is in the scale between the drift and the distance traveled, while b describes the trend of the drift with respect to the distance.

4. Experiments

Experiments are conducted to illustrate the validation of the established drift model. Moreover, some important facts of the drift in visual odometry are also revealed from the experimental results. We report about experiments which use simulated data, as well as a real data set (see Fig. 7). For both of these data, feature matching and tracking are simulated as in [1]. The reason for using simulated features is that visual odometry will not be affected in this case by other error sources except the controlled feature localization uncertainty, and will reveal the intrinsic properties of the drift.

We simulate stereo pairs of images. Feature-based visual odometry algorithms are considered, as they are more general compared to appearance-based algorithms and direct visual odometry, and it is easy to control them by the errors in the estimated motion matrix (by controlling feature localization uncertainties). No feature matching and tracking failures are considered in the simulation, thus robust regression is not adopted to remove the outliers.

Two typical feature-based visual odometry algorithms are implemented here, to illustrate the behavior of drift. The first one, named ABSOLUTE here, estimates the motion matrix between frames as an absolute orientation problem. Motion matrices are directly concatenated to estimate camera poses, so drift will not be suppressed, and is expected to be large. While the second one, named SBA here, uses sliding window sparse bundle adjustment to optimize the motion matrices as estimated by the ABSOLUTE method.

The whole implementation is similar as reported in [11]. The number of features tracked for both algorithms are set to 200, and the standard deviation of feature localization uncertainty is 0.5 pixel.

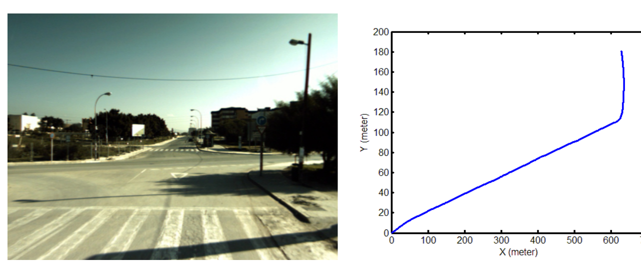

Figure 7. A frame from the real sequence, and the trajectory of the vehicle.

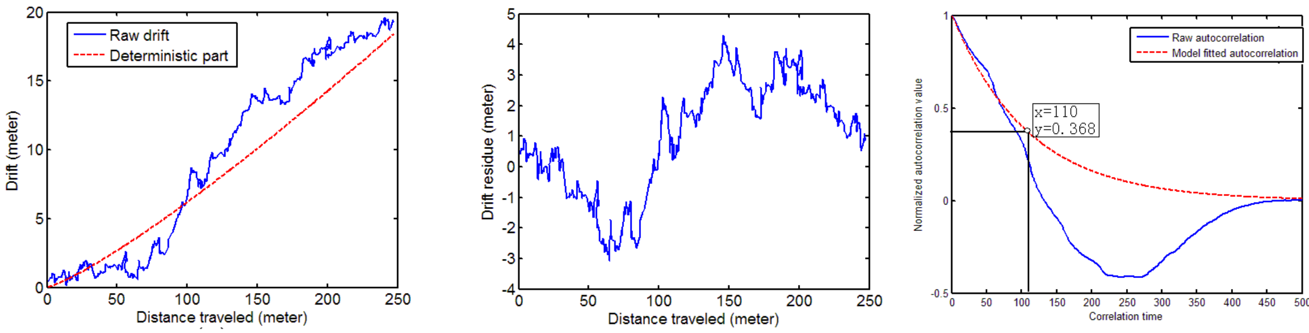

Figure 3. Experimental results for the ABSOLUTE method with simulated data. (left) Raw drift values and the deterministic part of the drift model with respect to the distance traveled when running the ABSOLUTE method for 500 steps. (middle) The residual ($\bar{u}_t$-process) component of the drift value. (right) Raw and model fitted autocorrelation values for the $\bar{u}_t$-process.

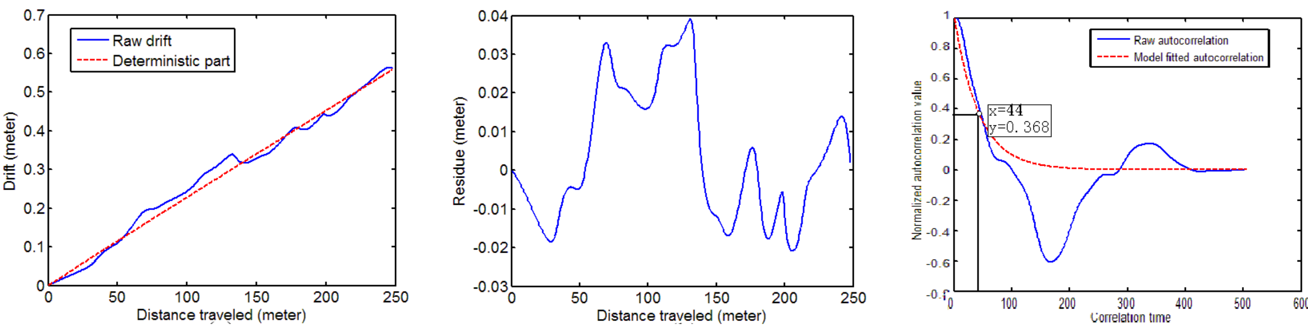

Figure 4. Experimental results for the SBA method with simulated data. Meaning for (left), (middle), and (right) as in Fig. 4.

For SBA, the number of frames for bundle adjustment equals 5. We assume that the frames are taken with a time interval of one second.

The results of the ABSOLUTE and SBA visual odometry algorithms for the simulated data are presented in Figs. 3 and 4, respectively.

The results with the real sequence from the Malaga datasets, as described in [13], are presented in Figs. 5 and 6. It can be seen that the absolute value of drift for SBA is much smaller than that for ABSOLUTE. But the distance-varying trend looks similar. For the identified parameters, as described in Tab. 1, e^a is much larger in ABSOLUTE than in SBA.

5. Discussion and conclusions

Modeling and analyzing long-range drift in visual odometry is of practical and theoretical significance. Drift in visual odometry is represented using an unbounded system model, and its analysis is divided into three steps, namely:

1. Estimating the distance-varying trends.

2. Computing a sample function of the residual process.

3. Characterizing the residual as a first-order Gauss-Markov process.

Experimental results reveal several important facts:

1. Modeling drift using an unbounded system model in form of a combination of a deterministic part and a first-order Gauss-Markov process is validated.

2. Drift in visual odometry will increase exponentially with the distance traveled, but different algorithms will increase differently.

3. Quantifying drift from a specific algorithm by τ and σ_u^2 is a more reasonable way than the usual offset ratio method.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant 50875169.

References

[1] H. Badino. Binocular ego-motion estimation for automotive applications. PhD thesis, Goethe University, Frankfurt, 2008. 4

[2] Y. Cheng, M. W. Maimone, and L. Matthies. Visual odometry on the Mars exploration rovers. *IEEE Robotics Automation Magazine*, 13(2):54–62, 2006. 1

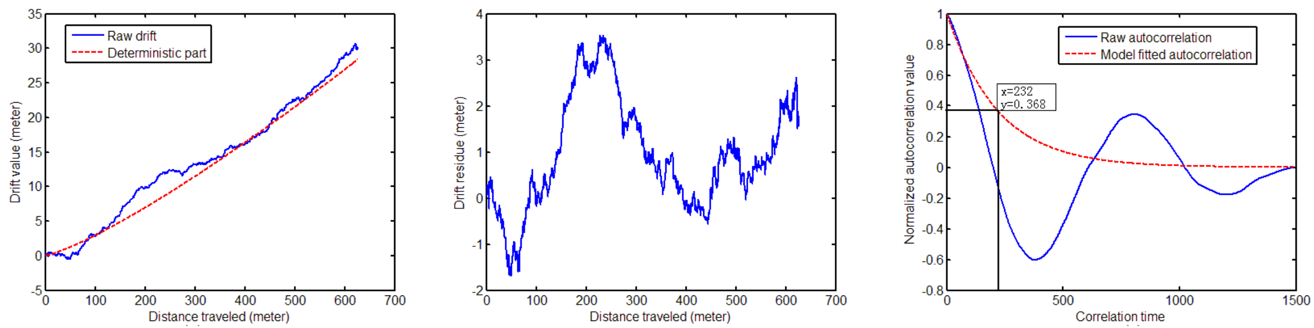

Figure 5. Experimental results for the ABSOLUTE method with the real sequence. Meaning for (left), (middle), and (right) as in Fig. 4.

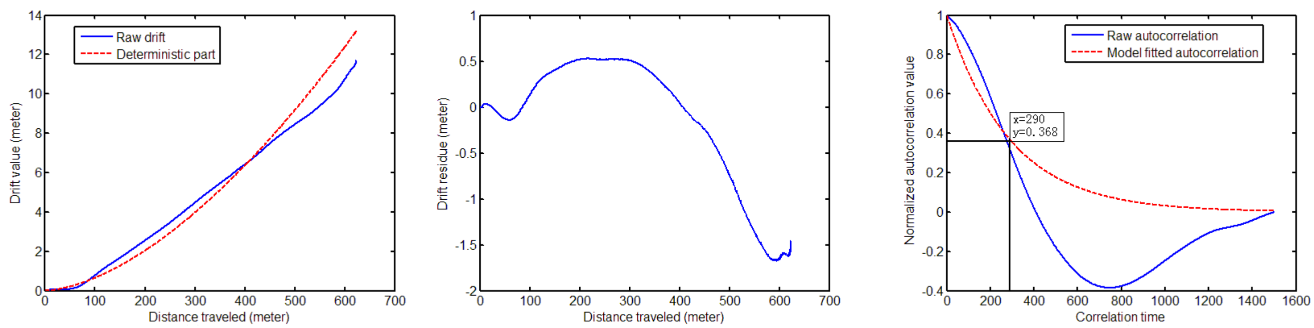

Figure 6. Experimental results for the SBA method with the real sequence. Meaning for (left), (middle), and (right) as in Fig. 4.

[3] A. Comport, E. Malis, and P. Rives. Real-time quadrifocal visual odometry. *Int. J. Robotics Research*, 29:245–266, 2010. 1

[4] G. Dubbelman and F. C. A. Groen. Bias reduction for stereo motion estimation with applications to large scale visual odometry. In *IEEE Conf. Computer Vision Pattern Recognition*, pages 2222–2229, 2009. 1

[5] J. Kelly and G. S. Sukhatme. An experimental study of aerial stereo visual odometry. In *IFAC Symp. Intelligent Autonomous Vehicles*, 2007. 1

[6] T. S. Lee and J. A. D'Appolito. On model identification of a class of unbounded systems with application to navigation instrument error modeling. *IEEE Trans. Automatic Control*, (2):470–474, 1981. 3, 4

[7] D. Nistér, O. Naroditsky, and J. Bergen. Visual odometry. In *IEEE Conf. Computer Vision Pattern Recognition*, volume 1, pages 652–659, 2004. 1, 2

[8] C. Olson, L. Matthies, M. Schoppers, and M. W. Maimone. Stereo ego-motion improvements for robust rover navigation. In *IEEE Int. Conf. Robotics and Automation*, pages 1099–1104, 2001. 1, 2

[9] C. Peter, D. Carrick, D. Matthew, H. Michael, R. Daniela, and V. Luliu. Experiments with underwater robot localization and tracking. In *IEEE Int. Conf. Robotics and Automation*, Roma, Italy, 2007. 1

[10] S. R. Scaramuzza Davide. Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles. *IEEE Trans. Robotics*, 24(5):1015–1026, 2008. 1

[11] N. Sünderhauf and P. Protzel. Towards using sparse bundle adjustment for robust stereo odometry in outdoor terrain. In *Towards Autonomous Robotic Systems*, pages 206–213, Guildford, UK, 2006. 1, 2, 4

[12] J.-P. Tardif, Y. Pavlidis, and K. Daniilidis. Monocular visual odometry in urban environments using an omnidirectional camera. In *IEEE/RSJ Int. Conf. Intelligent Robots Systems*, 2008. 1

[13] J. L. Blanco, F. A. Moreno, and J. Gonzalez. A collection of outdoor robotic datasets with centimeter-accuracy ground truth. *Autonomous Robots*, 27:327–351, 2009 5