CLAWAR 2018: 21st International Conference on Climbing and
Walking Robots and the Support Technologies for Mobile Machines,
Panama City, Panama, 10-12 September 2018

REAL TIME CAMERA HOMOGRAPHY ERROR MODELLING FOR SENSORIAL DATA FUSION[*]

RONY CABALLERO[†]

Faculty of Electrical Engineering, Universidad Tecnológica de Panamá, Campus Dr. Victor Levi Sasso, P.O.Box 0819-07289, Panama, Republic of Panama

ARANZAZU BERBEY

Faculty of Electrical Engineering, Universidad Tecnológica de Panamá, Campus Dr. Victor Levi Sasso, P.O.Box 0819-07289, Panama, Republic of Panama

The camera homography estimation involves using complex algorithms. They provide good results after some interactions, however the error model provided for such algorithm seems to be no appropriate for sensorial data fusion in real time. In this work a new methodology is proposed for camera homography and IMU data fusion. The experiment consists in rotate the camera. The experiment is done for frames. Here it is presented results about the error in homography estimation with and without bias compensation. The last section concerns about the experimental results and conclusions about this approach.

Keywords: camera homography estimation, IMU data fusion, bias compensation.

1. Introduction

In the last decades there has been remarkable progress in new robotics applications [1]–[4]. Among the design requirements for such kind of robots, there has been an important effort in using low cost sensor technologies. Now days it is possible to acquire IMUs and cameras for a few dollars. Nevertheless, such kind of Microelectromechanical Systems demands, drift, nonlinear or thermal compensation.

There are several approaches for inertial sensors drift and thermal compensation. However, a more effective one implementation applies first a bias, scale and misalignment effects compensation offline in addition with a real time drift correction. Drift correction could be realized by using additional measurement systems [5], [6].

[*] This work is supported by Secretaria Nacional de Ciencia, Tecnología e Innovación (SENACYT) of the government of Republic of Panama.

[†] Work partially supported by Universidad Tecnológica de Panamá.

Cameras are very useful sensors in many robotics applications. However, in order to obtain accurate measurement most of the cheaper ones, not only need calibration but also distortion compensation [7]–[9]. In addition, in order to track correctly image features, it is necessary to estimate the homography between consecutive images.

Camera homography estimation involves using complex algorithms [9-10]. Even though, these algorithms provide good results after some interactions, the error model provided for such algorithm seems to be no appropriate for sensorial data fusion in real time.

Therefore, it is necessary to estimate an appropriate homography error in real time for data fusion. In this work a new methodology is proposed for camera homography and IMU data fusion.

2. IMU and Camera Modelling

2.1. *Inertial Measurement Unit Error Modelling*

Inertial Measurement Units are usually formed by an array of three gyrorates and three accelerometers. Such kind of sensors present bias drift, scale and misalignment errors. These errors could be characterized approximately by the following equations [6],

$$\xi_\omega = B_\omega + N_\omega \alpha + M_\omega \omega + v_\omega \tag{1}$$

$$\xi_\alpha = B_\alpha + M_\alpha \alpha + v_\alpha \tag{2}$$

Where, $\alpha = \begin{bmatrix} \alpha_x & \alpha_y & \alpha_z \end{bmatrix}^T$ are the accelerations in IMU frame,

$\omega = \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix}^T$ are the rotational velocities in IMU frame,

$B_\omega = \begin{bmatrix} B_{\omega x} & B_{\omega y} & B_{\omega z} \end{bmatrix}^T$ are the gyrotates bias,

$B_\alpha = \begin{bmatrix} B_{\alpha x} & B_{\alpha y} & B_{\alpha z} \end{bmatrix}^T$ are the accelerometers bias,

$v_\omega = \begin{bmatrix} v_{\omega x} & v_{\omega y} & v_{\omega z} \end{bmatrix}^T$ are gyrorates white noise,

$v_\alpha = \begin{bmatrix} v_{\alpha x} & v_{\alpha y} & v_{\alpha z} \end{bmatrix}^T$ are accelerometers white noise,

$\xi_\omega = \begin{bmatrix} \xi_{\omega x} & \xi_{\omega y} & \xi_{\omega z} \end{bmatrix}^T$ and $\xi_\alpha = \begin{bmatrix} \xi_{\alpha x} & \xi_{\alpha y} & \xi_{\alpha z} \end{bmatrix}^T$ represent the gyrorates and accelerometers errors in IMU frame respectively. Additionally,

N_ω is a matrix that considers the acceleration effects in gyrorates outputs, M_ω and M_α are matrices that consider scale and misalignment errors in IMU unit.

There are some calibration algorithms that consider bias, scale and misalignment compensation offline. Nevertheless, it will be very useful if a real time drift correction is implemented. Such kind of correction could be realized by using GPS, electronic compass, active ranging sensors or cameras.

2.2. *Camera Model*

There are several image camera models that could be applied. However, this work will be limited for such applications where scene depth is small compared to average distance to camera. Therefore, the pinhole camera model is chosen to map any visible point in the fixed reference frame (x,y,z) to image plane (u,v).

2.2.1. *Pinhole Camera Model*

In this approximation, the origin of camera reference frame matches with optical center and z axis matches with camera optical axis. In addition, the image plane (u,v) is not only parallel to (x,y) plane, but also u and v axis are parallel to x and y axis respectively.

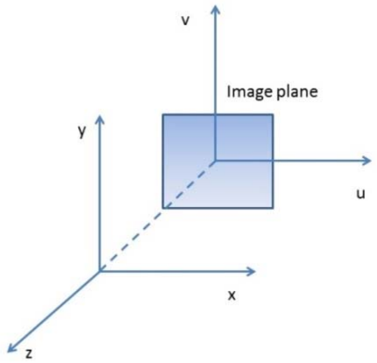

Figure 1. Pinhole Camera Model. Coordinates reference frames.

Any point in reference frame (x,y,z) could be mapped to image plane (u,v) by the following approximation,

$$u \approx \frac{k_x}{z} x \left(1 + \varepsilon \left(x^2 + y^2\right)\right) \tag{3}$$

$$v \approx \frac{k_y}{z} y \left(1 + \varepsilon \left(x^2 + y^2\right)\right) \tag{4}$$

Where, k_x and k_y are the effective focal lengths of the camera. ε is a parameter that approximates radial optical distortion. If ε parameter is positive the image suffers a pincushion effect, but, if it is negative the image suffers a barrel effect. The parameters k_x and k_y could be obtained using an appropriate calibration. Then, radial distortion could be compensated by using Fitzgibbons approach [6],

$$u_{CF} = \frac{u}{1+\varepsilon_{CF}\left(u^2+v^2\right)} \tag{5}$$

$$v_{CF} = \frac{v}{1+\varepsilon_{CF}\left(u^2+v^2\right)} \tag{6}$$

Where, ε_{CF} is an appropriate chosen parameter that compensates radial optical distortion.

2.2.2. *Affine Transformation*

In order to fusion IMU and camera data is necessary to track image features in video frames. Therefore, it is necessary to analyze how two consecutive frames changes. This relation between two images could be modeled with affine transformation,

$$\begin{bmatrix} u_2 \\ v_2 \\ 1 \end{bmatrix} = H \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} \tag{7}$$

Where, $\left(u_1, v_1\right)$ corresponds to pixel coordinates from first image, $\left(u_2, v_2\right)$ corresponds to pixel coordinates from second image and H is the homography matrix. The affine transformation homography matrix has the following form,

$$H = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \tag{8}$$

Where, *c* and *f* reflect *u* and *v* translations respectively. In other hand, *a, b, d* and *e* are used to model rotation, shear and scale changes.

2.2.3. *Homography Matrix Estimation*

The homography matrix estimation is a complex process. Various algorithms have been proposed in order to estimate the homograpy matrix between two images. However, must of them consider the following steps: feature detection, feature matching and fitting matrix parameters [10]. The most effective image features for feature matching are corners and SIFTs (Scale Invariant Features Transform). The feature matching could be implemented with the Scott and Higgins Features Association Algorithm [11]. On the other hand, the matrix parameters could be fitted with RANSAC algorithm (Random Sampling and Consensus Algorithm)[12].

Even though, the RANSAC algorithm is a powerful tool to estimate homography matrix, it does not ensure an unbiased solution. In addition, it does not provide any information about the variances of matrix parameters. Such information could be very useful for sensorial data fusion.

3. Homography Matrix Statistics

The uncertainty around homography matrix parameters could be obtained with the help of feature matching matrix. Even though, this matrix was used by RANSAC algorithm to find homography matrix parameters, it contains outliers data points. These outliers could be produced by low contrast, non-uniform lighting and blurring effects.

3.1. *Homography Error Model Estimation.*

3.1.1. *Bias and Variance Estimation of parameter c and f*

For n samples, c and f bias could be estimated with,

$$\begin{bmatrix} \delta c_k \\ \delta f_k \end{bmatrix} = \frac{1}{n}\sum_{k=1}^{n}\begin{bmatrix} u_{2k} \\ v_{2k} \end{bmatrix} - \begin{bmatrix} \hat{a} & \hat{b} \\ \hat{d} & \hat{e} \end{bmatrix}\begin{bmatrix} u_{1k} \\ v_{1k} \end{bmatrix} - \begin{bmatrix} \hat{c} \\ \hat{f} \end{bmatrix} \tag{9}$$

And variances are estimated with,

$$\begin{bmatrix} \sigma_{ck}^2 \\ \sigma_{fk}^2 \end{bmatrix} = \frac{1}{n-1}\sum_{k=1}^{n}\begin{bmatrix} \left(u_{2k} - \hat{a}u_{1k} - \hat{b}v_{1k} - \hat{c} - \delta c\right)^2 \\ \left(v_{2k} - \hat{d}u_{1k} - \hat{e}v_{1k} - \hat{f} - \delta f\right)^2 \end{bmatrix} \tag{10}$$

3.1.2. *Bias and Variance Estimation of parameter a and d*

For n samples, a and d bias could be estimated with,

$$\begin{bmatrix} \delta a_k \\ \delta d_k \end{bmatrix} = \frac{1}{n}\sum_{k=1}^{n}\begin{bmatrix} \dfrac{u_{2k} - \hat{b}v_{1k} - \hat{c} - \delta c}{u_{1k}} - \hat{a} \\ \dfrac{v_{2k} - \hat{e}v_{1k} - \hat{f} - \delta f}{u_{1k}} - \hat{d} \end{bmatrix} \tag{11}$$

And variances are estimated with,

$$\begin{bmatrix}\sigma_{ak}^2\\ \sigma_{dk}^2\end{bmatrix}=\frac{1}{n-1}\sum_{k=1}^{n}\begin{bmatrix}\left(\dfrac{u_{2k}-\hat{b}v_{1k}-\hat{c}-\delta c}{u_{1k}}-\hat{a}-\delta a\right)^2\\ \left(\dfrac{v_{2k}-\hat{e}v_{1k}-\hat{f}-\delta f}{u_{1k}}-\hat{d}-\delta d\right)^2\end{bmatrix} \tag{12}$$

3.1.3. *Bias and Variance Estimation of parameter b and e*

For n samples, b and e bias could be estimated with,

$$\begin{bmatrix}\delta b_k\\ \delta e_k\end{bmatrix}=\frac{1}{n}\sum_{k=1}^{n}\begin{bmatrix}\dfrac{u_{2k}-\left(\hat{a}+\delta a\right)u_{1k}-\hat{c}-\delta c}{v_{1k}}-\hat{b}\\ \dfrac{v_{2k}-\left(\hat{d}+\delta d\right)u_{1k}-\hat{f}-\delta f}{v_{1k}}-\hat{e}\end{bmatrix} \tag{13}$$

And variances are estimated with,

$$\begin{bmatrix}\sigma_{bk}^2\\ \sigma_{ek}^2\end{bmatrix}=\frac{1}{n-1}\sum_{k=1}^{n}\begin{bmatrix}\left(\dfrac{u_{2k}-\left(\hat{a}+\delta a\right)u_{1k}-\hat{c}-\delta c}{v_{1k}}-\hat{b}-\delta b\right)^2\\ \left(\dfrac{v_{2k}-\left(\hat{d}+\delta d\right)u_{1k}-\hat{f}-\delta f}{v_{1k}}-\hat{e}-\delta e\right)^2\end{bmatrix} \tag{14}$$

3.2. *Translation, Rotation and Scale Estimation.*

The parameters and variances calculated before could be used to estimate any change in translation, rotation or scale. This information could be used for sensor fusion.

3.2.1. *Translation Estimation*

A change in image translation is directly related with c and f parameters,

$$\begin{bmatrix}\Delta u\\ \Delta v\end{bmatrix}=\begin{bmatrix}c\\ f\end{bmatrix} \tag{15}$$

The variance could be approximated as follows,

$$\begin{bmatrix}\sigma_u^2\\ \sigma_v^2\end{bmatrix}=\begin{bmatrix}\sigma_c^2\\ \sigma_f^2\end{bmatrix} \tag{16}$$

3.2.2. *Rotation Estimation*

A change in image rotation could be calculated with a and b parameters,

$$\Delta\theta_z=\operatorname{atan}\left(\frac{\hat{b}+\delta b}{\hat{a}+\delta a}\right) \tag{17}$$

Meanwhile, if rotation is small, the variance could be approximated as follows,

$$\sigma_{\theta z}^2\approx\left(\frac{1}{\hat{a}^2}\right)\sigma_b^2+\left(\frac{\hat{b}^2}{\hat{a}^4}\right)\sigma_a^2 \tag{18}$$

3.2.3. *Scale Estimation*

Any change in image scale could be calculated with,

$$s_u=\sqrt{\hat{a}^2+\hat{b}^2} \tag{19}$$

variance could be approximated as follows,

$$\sigma_{su}^2=\left(\frac{4\hat{a}^2}{\hat{a}^2+\hat{b}^2}\right)\sigma_a^2+\left(\frac{4\hat{b}^2}{\hat{a}^2+\hat{b}^2}\right)\sigma_b^2 \tag{20}$$

4. Data Fusion Methodology

It is important to stablish a relationship between IMU and camera data. The camera homography could be useful to calculate inertial sensors bias if the IMU and camera reference frame are the same. Therefore, any change in the image plane is related with changes in IMU frame,

$$\begin{bmatrix}\Delta u\\ \Delta v\\ \Delta\theta_z\end{bmatrix}=\begin{bmatrix}\frac{k_x}{z} & 0 & 0 & \frac{k_x l_z}{z} & 0\\ 0 & \frac{k_y}{z} & -\frac{k_y l_z}{z} & 0 & 0\\ 0 & 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\Delta x\\ \Delta y\\ \Delta\theta_x\\ \Delta\theta_y\\ \Delta\theta_z\end{bmatrix} \tag{21}$$

Where, l_z is the average distance from camera reference frame origin to scene plane. This last equation (21) could be used for data fusion. However, accelerometers and gyrorates bias drift correction needs additional measurement systems to be estimated simultaneously.

One possible solution is based that in some applications translations and rotations movements are done separately. Therefore, bias correction to accelerometers or gyrorates could be applied independently.

4.1. *Accelerometer Bias Correction.*

The accelerometer bias correction could be estimated as follows,

$$\begin{bmatrix} b_{axk} \\ b_{ayk} \end{bmatrix} = \begin{bmatrix} \dfrac{1}{n-m} \sum_{i=m+1}^{n} \dfrac{z_i c_i}{m T^2 k_x} - \dfrac{z_{i-m} c_{i-m}}{m T^2 k_x} - a_{xi} \\ \dfrac{1}{n-m} \sum_{i=m+1}^{n} \dfrac{z_i f_i}{m T^2 k_y} - \dfrac{z_{i-m} f_{i-m}}{m T^2 k_y} - a_{yi} \end{bmatrix} \tag{22}$$

Where, a_{xi} and a_{yi} are accelerometers outputs, m is a parameter that adjusts acceleration estimation average.

4.2. *Gyrorate Bias Correction.*

The gyrorate bias correction could be estimated as follows,

$$\begin{bmatrix} b_{\omega xk} \\ b_{\omega yk} \\ b_{\omega zk} \end{bmatrix} = \begin{bmatrix} \dfrac{1}{n} \sum_{i=1}^{n} - \dfrac{z_i f_i}{T k_y l_{zi}} - \Omega_{xi} \\ \dfrac{1}{n} \sum_{i=1}^{n} \dfrac{z_i c_i}{T k_x l_{zi}} - \Omega_{yi} \\ \dfrac{1}{n} \sum_{i=1}^{n} \dfrac{\Delta \theta_{zi}}{T} - \Omega_{zi} \end{bmatrix} \tag{23}$$

Where, Ω_{xi}, Ω_{yi} and Ω_{zi} are gyrorates outputs.

5. Experimental Results

The proposed methodology approach is tested for estimating the heading angle of camera. The algorithm uses the Scott and Higgins Features Association, RANSAC and a Kalman filter algorithms to fuse camera and IMU data.

5.1. *Experimental Setup.*

The utilized hardware consist in a color video camera 240x320 and a IMU MPU6050 connected to a Pentium i5 using a frame grabber and a Raspberry PI 3 respectively. The sampling rates are chosen to operate the camera at to 2 images per second and the IMU to 20 samples per second.

5.2. *Kalman Filtering.*

The Kalman filter (please see figure 2) is tuned with the following parameters,

$$x_k = \begin{bmatrix} \theta_k & \dot{\theta}_k & b_k \end{bmatrix}^T, \qquad u_k = 0,\ \Gamma_k = 0,\ \Lambda_k = 1,$$

$$y_k = \begin{bmatrix} \Omega_k - b_{\omega yk} & \dfrac{z_k c_k}{l_z k_x} \end{bmatrix}^T, \qquad \frac{z_k}{l_z k_x} \approx 0.2026,$$

$$\Phi_k = \begin{bmatrix} 1 & T & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad Q_k = \begin{bmatrix} 0.0031 & 0 & 0 \\ 0 & 0.0627 & 0 \\ 0 & 0 & 0.0003 \end{bmatrix},$$

$$H_k = \begin{cases} \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix} & \text{for } k = 10, 20, 30, 40, 50, \ldots \\ \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} & \text{for } k \neq 10, 20, 30, 40, 50, \ldots \end{cases}, \qquad R_k = \begin{bmatrix} 0.33 & 0 \\ 0 & \sigma_{ck}^2 \end{bmatrix}$$

Where, θ is heading angle, $\dot{\theta}$ is the heading angle rate, *b* is gyrorate bias and Ω_k is the gyrorate output.

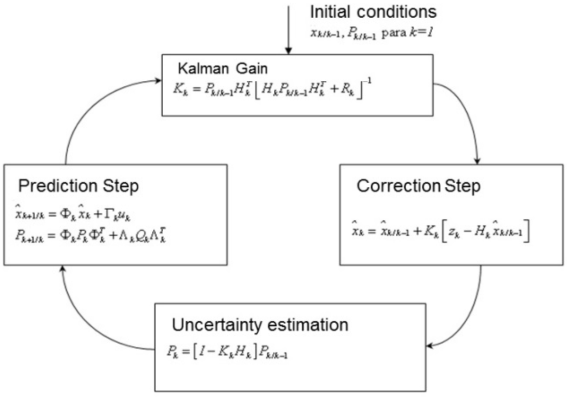
Initial conditions
$x_{k/k-1}, P_{k/k-1}$ para $k=1$
Kalman Gain
$K_k = P_{k/k-1} H_k^T \left[H_k P_{k/k-1} H_k^T + R_k \right]^{-1}$
Prediction Step
$\hat{x}_{k+1/k} = \Phi_k \hat{x}_k + \Gamma_k u_k$
$P_{k+1/k} = \Phi_k P_k \Phi_k^T + \Lambda_k Q_k \Lambda_k^T$
Correction Step
$\hat{x}_k = \hat{x}_{k/k-1} + K_k \left[z_k - H_k \hat{x}_{k/k-1} \right]$
Uncertainty estimation
$P_k = \left[I - K_k H_k \right] P_{k/k-1}$

Figure 2. Kalman Filter Algorithm.

5.3. *Result Analysis.*

The experiment consists in rotate de camera 30 degrees. The experiment is done for 14 frames (See figure 3). The homography is first calculated for each pair of frames. Then, homography bias is corrected with help of equation (9). The root mean square error homography is shown in the table 1 for uncompensated and compensated homography. It is clear that camera provides more accurate estimation, however it is slower than gyrorates. Now, the Kalman filter is used for fuse camera and gyrorate information (figure 4). Here, gyrorate data is available for each filter correction step, but camera data is only available each ten steps.

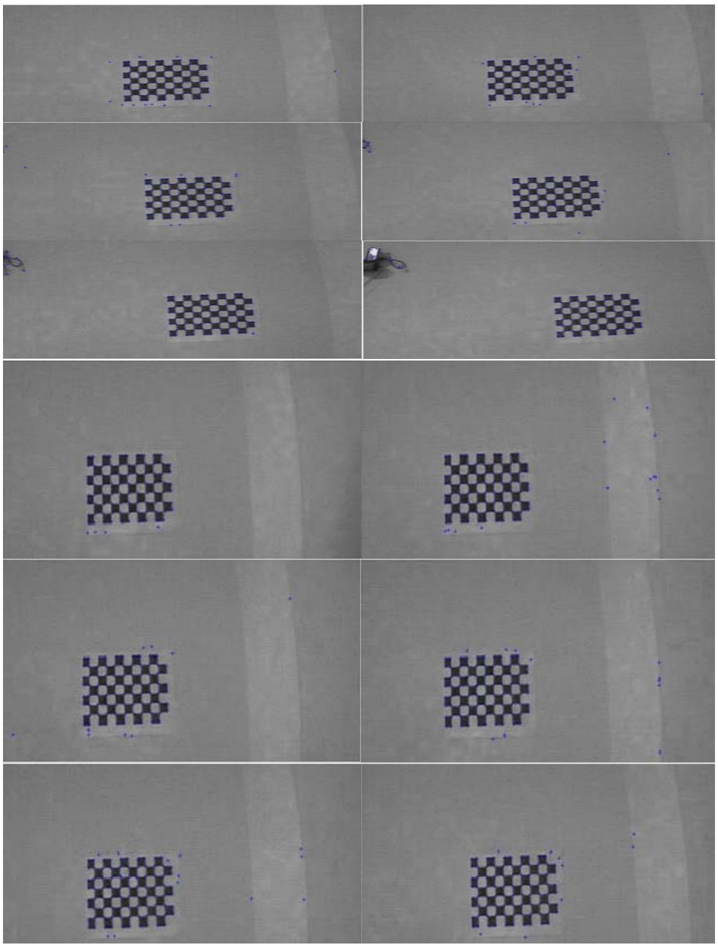

Figure 3. Frames with corners detection.

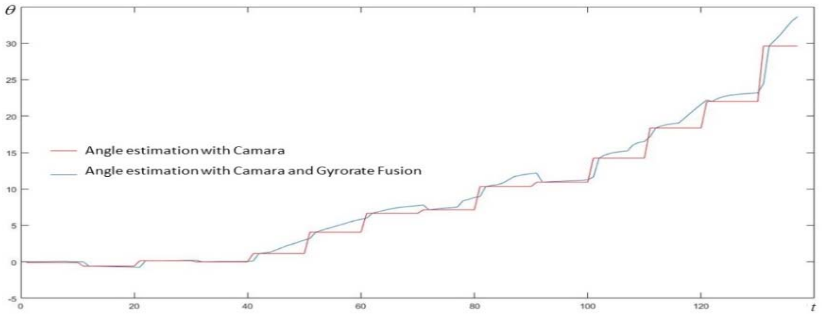

Figure 4. Heading angle estimation.

Table 1: Root mean square error in homography estimation for 800 interactions RANSAC with and without bias compensation.

	Frames 0 to 1	Frames 1 to 2	Frames 2 to 3	Frames 3 to 4	Frames 4 to 5	Frames 4 to 6	Frames 6 to 7
without compensation	1.1551	0.9389	4.3063	0.2882	2.7688	2.8641	4.0229
with compensation	0.9295	0.5755	0.9494	0.2877	0.9075	0.5464	2.5672
	Frames 7 to 8	Frames 8 to 9	Frames 9 to 10	Frames 10 to 11	Frames 11 to 12	Frames 12 to 13	Frames 13 to 14
without compensation	1.8736	0.7202	0.4909	0.7146	2.0290	2.8287	4.3092
with compensation	1.5800	0.3466	0.1091	0.4921	2.0158	2.4149	3.1275

6. Conclusions

The homography error model estimation was not only a useful tool for data fusion, but also for improving homography. The bias compensation improves system velocity, since RANSAC interaction could be diminished. However, in this aspect, more research must be done in order to obtain an appropriate value of interactions. Also, lot of further work is necessary in order to improve data fusion methodology. It is interesting, to analyze fusion with two or more cameras, other sensors technologies or applying artificial intelligence algorithm approaches.

Acknowledgments

The authors want to express their gratitude to the National Secretary of Science and Technology (SENACYT) of the Government of the Republic of Panama and Technological University of Panama (UTP).

References

1. M. Quan, S. Piao, M. Tan, and S.-S. Huang, “Accurate Monocular Visual-inertial SLAM using a Map-assisted EKF Approach,” pp. 1–12, 2017.
2. F. M. Mirzaei and S. I. Roumeliotis, “A Kalman Filter-Based Algorithm for IMU-Camera Calibration: Observability Analysis and Performance Evaluation,” *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1143–1156, 2008.
3. K. Nirmal, A. G. Sreejith, J. Mathew, M. Sarpotdar, A. Suresh, A. Prakash, M. Safonova, and J. Murthy, “Noise modeling and analysis of an IMU-based attitude sensor: improvement of performance by filtering and sensor fusion,” 2016.
4. M. Alatise and G. Hancke, “Pose Estimation of a Mobile Robot Based on Fusion of IMU Data and Vision Data Using an Extended Kalman Filter,” *Sensors*, vol. 17, no. 10, p. 2164, 2017.
5. S. Romaniuk and Z. Gosiewski, “Kalman filter realization for orientation and position estimation on dedicated processor,” *Acta Mech. Autom.*, vol. 8, no. 2, pp. 88–94, 2014.
6. E. H. Shin and N. El-Sheimy, “A new calibration method for strapdown inertial navigation systems,” *Zeitschrift für Vermessungswes.*, vol. 127, no. 1, pp. 41–50, 2002.
7. J.Y.Weng, P.Cohen, and M.Herniou, “Camera calibration with distortion model and accuracy evaluation,” *IEEE Trans. Pattern Anal. Mach. Intell*, vol. 14, no. 10, pp. 965–980, 1992.
8. J. H. Brito, R. Angst, K. Koser, and M. Pollefeys, “Radial distortion self-calibration,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 1368–1375, 2013.
9. L. Ma, Y. Chen, and K. L. Moore, “A Family of Simplified Geometric Distortion Models for Camera Calibration,” vol. 1, no. 2, pp. 1–14, 2003.
10. R. Hartley and A. Zisserman, “Multiple View Geometry in Computer Vision Second Edition,” *CEUR Workshop Proc.*, vol. 1542, no. 9, pp. 33–36, 2015.
11. G. L. Scott and H. C. Longuet-Higgins, “An Algorithm for Associating the Features of Two Images,” *Proc. R. Soc. B Biol. Sci.*, vol. 244, no. 1309, pp. 21–26, 1991.
12. I. Austvoll and B. Kwolek, “39.Region covariance matrix-based object tracking with occlusions handling.pdf,” pp. 1–8.