Comparision of computer vision and photogrammetric approaches for motion estimation of object in an image sequence

Tserennadmid Tumurbaatar
Department of Information and Computer Science
National University of Mongolia
Ulaanbaatar, Mongolia
tserennadmid@seas.num.edu.mn

Taejung Kim
Department of Geoinformatic Engineering
Inha University
Incheon, Korea
tezid@inha.ac.kr

***Abstract*—3D tracking plays a vital role in 3D applications by enhancing interaction between real and virtual world. We present various real-time 3D motion estimation approaches developed in photogrammetry and computer vision fields and compare their performance. The methods developed in both fields estimates 3D motion of a moving object relative to a camera or equivalently moving camera relative to the object in an image sequence when its corresponding features are known at different times. We reviewed 3D motion models formulated by different methods related to their geometric properties. We implemented four different methods and analyzed their performance results. Comparison from test datasets from image sequences demonstrated that homography based approaches in both fields were more accurate than relative orientation or essential matrix based approaches under noisy situations.**

Keywords-motion estimation; correspondence point; moving object;

I. INTRODUCTION

Recovering the motion of the object from an image sequence is an important task in variety of applications, including augmented reality, 3D navigation and manipulation. The motion estimation problem has been proposed under different approaches depending on the image sensor and choice of methodology.

In this study, we present real-time motion estimations of a moving object in image sequences taken by a single camera. We have used four estimation methods for determining 3D motion based on tracked feature correspondences. Two of them are developed under computer vision approach, and two of them under photogrammetric approach. We note that most previous investigations for 3D motion estimation have not compared the methods developed in computer vision and photogrammetry thoroughly. Since a number of applications have evolved by linking techniques developed in both fields in recent years, we aim to contribute to this motivation. We will point out their theoretical and practical differences at implementation level.

In computer vision, automatic relative orientation of image sequences has been widely investigated with assumption of a calibrated camera. Essential matrix is defined up to a scale factor for translations as set of linear homogeneous equations by establishing feature correspondences. Relative pose parameters of the perspective two views are computed by decomposing an essential matrix [1]-[2]. The estimation methods of relative camera pose from various number of point correspondences in various applications were developed [3]-[5]. Moreover, pose parameters of a camera relative to planar object can be estimated by decomposing a homography matrix through point correspondences. The numerical and analytical methods for 3D pose from homography decomposition were introduced in detail [6]. The authors in [7]-[10] also introduced 3D pose estimation based on homography matrix in augmented reality and robot control applications.

In classic photogrammetry, to determine the position and orientation of right image relative to left image frame from tie-points by assuming known intrinsic parameters is an important task where no ground truth is assumed. It is also well known as relative orientation process if a sufficient set of corresponding points in an image sequence have been identified [11]. Mathematically, relative orientation parameters as motion parameters can be determined by collinear or coplanar equations [12]-[15].

This paper organized as follows. Mathematical models for approaches will be described in Section 2. Methodology of implementation steps will be presented in Section 3. Performance analysis of estimations will be discussed in Section 4.

II. MOTION MODEL

We have assumed that a camera is stationary. The camera has taken an image sequence of the moving object through its field of view.

Let the coordinate system be fixed on the camera with its origin O at the optical center. The z axis is coinciding with the optical axis and pointing to the direction of view. Without loss of generality, we assume that focal length is unity. The image plane is located at a distance equal to the focal length. Consider $X_1 \in \mathbb{R}^3$ object space coordinates of a point P_1 on a rigid object moves to $X_2 \in \mathbb{R}^3$ object space coordinate of a point P_2 with respect to a camera coordinate system. Using a perspective projection model, the point P_1 is projected at $x_1 \in \mathbb{R}^3$ image space coordinates of a point p_1 at time t_1. Similarly, the point P_2 is projected at $x_2 \in \mathbb{R}^3$ image space coordinates of a point p_2 at time t_2 on the image plane as shown in Fig. 1.

To summarize, our problem is:

Given two image views with correspondences (p_1, p_2),
Find 3D rotation and 3D translation up to scale.

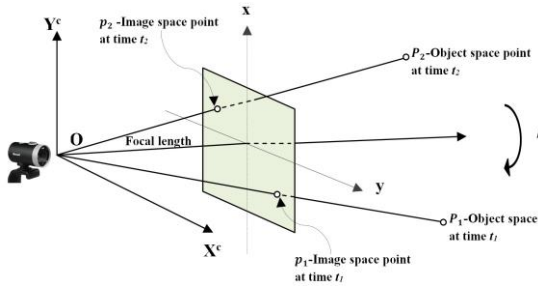

Figure 1. Geometry of imaging system.

Due to the rigidity constraint of the object motion, P_1 and P_2 are related by rotation matrix R and translational vector T:

$$X_2 = RX_1 + T \qquad (1)$$

This can be written in triple product of vector, which is often called as coplanarity or epipolar constraint.

$$x_2^T \hat{T} R x_1 = 0 \qquad (2)$$

Associated with Eq. (2), we will define proposed methods for motion estimation in both fields.

A. Recovering 3D motion from essential matrix in computer vision

We can reformulate Eq. (2) as well-known essential matrix for the relative pose between two views.

$$E = \hat{T}R \in \mathbb{R}^{3\times3}, \qquad x_2^T E x_1 = 0 \qquad (3)$$

Here, $\hat{T}$ is defined as $\hat{T} = \begin{bmatrix} 0 & -B_z & B_y \\ B_z & 0 & -B_x \\ B_y & B_x & 0 \end{bmatrix}$

E has singular value decomposition (SVD) as defined:

$$E = U\Sigma V^T \qquad (4)$$

where U and V are chosen such that $\det(U) > 0$ and $\det(V) > 0$, and $\Sigma = \text{diag}\{1, 1, 0\}$. Furthermore, the following formulae give the two distinct solutions for rotation and translation vector from essential matrix.

$$R = U \begin{bmatrix} 0 & \mp1 & 0 \\ \pm1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} V^T, \; \hat{T} = U \begin{bmatrix} 0 & \mp1 & 0 \\ \pm1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} U^T \qquad (5)$$

One of the four possible solutions corresponds true solution for Eq. (5) that can be chosen by enforcing constraint, called cheirality test [16].

B. Recovering 3D motion from homography matrix in computer vision

Let a point P_i on a 2D plane π in 3D space, $n = [n_1, n_2, n_3]$ be the unit normal vector to the plane π, and d $(d > 0)$ denote the distance from the plane π to the optical center of the camera. Suppose the optical center of the camera never passes through the plane π. Then we have as following equation from Eq. (1) with normalizing translational vector T by plane depth d:

$$X_2 = RX_1 + T = \left(R + \frac{1}{d}Tn^T\right)X_1 = HX_1 \qquad (6)$$

Here,

$H = \left(R + \frac{1}{d}Tn^T\right) \in \mathbb{R}^{3\times3}$

We call the matrix H as the planar homography matrix, since it denotes a linear transformation from $X_1 \in \mathbb{R}^3$ to $X_2 \in \mathbb{R}^3$. Since H depends on the motion parameters $\{R, T\}$ and the structure parameters $\{n, d\}$, due to scale ambiguity in the term $\frac{1}{d}T$ in equation (6), we have homography mapping induced by a plane π.

$$x_2 \sim Hx_1 \qquad (7)$$

After we have recovered H from at least four point correspondences, we can decompose such a matrix into its motion and structure parameters by SVD [6].

$$H = U\Sigma V^T \qquad (8)$$

where U and V are orthogonal matrices and a diagonal matrix is Σ, which contains singular value of H.

After decomposition, we also obtain four solutions: two completely different solutions and their opposites for decomposing H matrix. In order to reduce the number of physically possible solutions, we impose the positive depth constraint, having $n^T e > 0, e = [0,0,1]^T$, since the camera can see only points in front of it.

C. Recovering 3D motion from relative orientation in photogrammetry

In photogrammetry relative orientation is a well-known process of determining relative position and orientation of the first view of the frame with respect to next view of the frame in a sequence. As shown in Eq. (2), essential matrix determined in computer vision is mathematically identical to the equation determined in coplanar condition used in photogrammetry. This has been well confirmed in [14][16]. By reformulating equivalently non-linear equations in coplanar condition [13] to identical Eq. (2), we can write as following:

$$\begin{bmatrix} D_1 \\ E_1 \\ F_1 \end{bmatrix} = \begin{bmatrix} ix_1 \\ jx_1 \\ kx_1 \end{bmatrix}, \quad \begin{bmatrix} D_2 \\ E_2 \\ F_2 \end{bmatrix} = \begin{bmatrix} ix_2 \\ jx_2 \\ kx_2 \end{bmatrix},$$

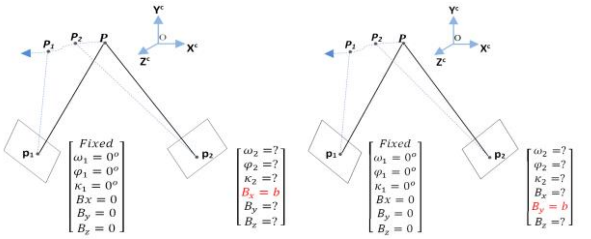

Figure 2. Parameter configurations of relative orientation

$$\mathrm{R} = \begin{bmatrix} i_x & j_x & k_x \\ i_y & j_y & k_y \\ i_z & j_z & k_z \end{bmatrix}, \quad b = [B_x, B_y, B_z] \tag{10}$$

$i_x = \cos\varphi * \cos\kappa;\ j_x = \cos\varphi * \sin\kappa;\ \ k_x = \sin\varphi;$
$i_y = \ \sin\omega * \sin\varphi * \cos\kappa + \ \cos\omega * \sin\kappa;$
$j_y = -\sin\omega * \sin\varphi * \sin\kappa \ + \ \cos\omega * \cos\kappa;$
$k_y = -\sin\omega * \cos\varphi;$
$i_z = -\cos\omega * \sin\varphi * \cos\kappa \ + \ \sin\omega * \sin\kappa;$
$j_z = \ \cos\omega * \sin\varphi * \sin\kappa \ + \ \sin\omega * \cos\kappa;$
$k_z = \cos\omega * \cos\varphi;$

Then triple-scalar product of the three vectors as following.

$$B_x \cdot (E_1F_2 - E_2F_1) + B_y \cdot (F_1D_2 - F_2D_1) + B_z \cdot (D_1E_2 - D_2E_1) = 0 \tag{11}$$

Here, ω is rotation about the x axis, φ is rotation about the y axis, and κ is rotation about the z axis. B_x is translation about the x axis, B_y is translation about the y axis, and B_z is translation about the z axis. From this non-linear equation the three unknown parameters of the rotation matrix R and the two unknown components of the base vector can be obtained by Taylor's linearization in a least squares solution. We set all six variables equal to zero for the first view by considering no movement of object in this view. The iterative method for solving non-linear equation requires an initial guess for each unknown parameter for its convergence. In particular, we can make the simplifying assumption that $\omega = \varphi = \kappa = 0^{\circ}$ for the second view orientation by assuming a constant fixed value for B_x, *or* B_y based on parallax differences between two views as illustrated in Fig. 2.

D. Recovering 3D motion from homography based relative orientation in photogrammetry

By reformulating matrix Eq. (7), we can obtain non-linear equation since the mapping from the first to the next image is given by the homography.

$$\mathrm{x}_2 \cong \left(\mathrm{R} + \frac{1}{\mathrm{d}}\mathrm{T}\mathrm{n}^{\mathrm{T}}\right)\mathrm{x}_1 \tag{12}$$

The eight unknown parameters of this equation can be described for the five parameters $(\omega, \varphi, \kappa, B_x, B_z)$ of the relative motion and three parameters (n_1, n_2, n_3) of the plane in object space. Similarly, this equation can be solved by Taylor's linearization in a least squares solution with a priori fixed value for B_x, *or* B_y, as illustrated in Fig. 2. Note that we set 1 for variable of n_3 for its initial value.

III. METHODOLOGY

In this section, we explain implementation steps of the proposed method for estimation of the motion parameters. Fig. 3 provides a process flow of the proposed method. We estimate 3D motion parameters of the moving object for two consecutive frames with point correspondences between them from a single camera.

Firstly, we capture an initial frame at time t_1, and extract a template region from it as described in Fig. 3. Then feature points are computed for the extracted template region by using SIFT feature extractor as illustrated in Fig. 3. Once we define template region in the initial frame, real-time processing is started with all computational steps.

Secondly, when a new frame at time t_2 of a moving object is captured, feature points of the new frame are extracted. Feature points of the new frame are matched with the feature points of the template region as described in Fig. 3. The best matches as corresponding points between two frames are found by a Brute Force matcher with eliminating outliers among the matched points based on RANSAC (Random Sample Consensus) method. To eliminate outliers in the feature correspondences before estimation of the motion parameters, we have used different RANSAC method for each of the four proposed approaches: homography-based RANSAC for the estimation method using homography decomposition in computer vision and the method using relative orientation based on homography in photogrammetry; essential matrix-based RANSAC for the method using the decomposition of essential matrix in computer vision; and relative orientation-based RANSAC for the method using the relative orientation parameters in photogrammetry.

Thirdly, when all processing steps are accumulated as mentioned above, motion estimations are implemented for the matched corresponding points between the initial frame and the next frames of the image sequences.

IV. PERFORMANCE OF 3D MOTION ESTIMATION

We implemented the proposed methods in photogrammetry and computer vision with the C++ programming language, Visual Studio programming tool, OpenCV 2.4.9 library and OpenGL graphic library on a PC with the Intel(R) Core(TM) i5, CPU 3.0 GHz, 4096 RAM and, with a Microsoft LifeCam. Intrinsic parameters of the camera such as focal length, principle point, and lens distortion coefficients are known by camera calibration toolbox, GML. We examined the performance of the four estimations by a real dataset created from real scenes and a simulated dataset created from OpenGL library.

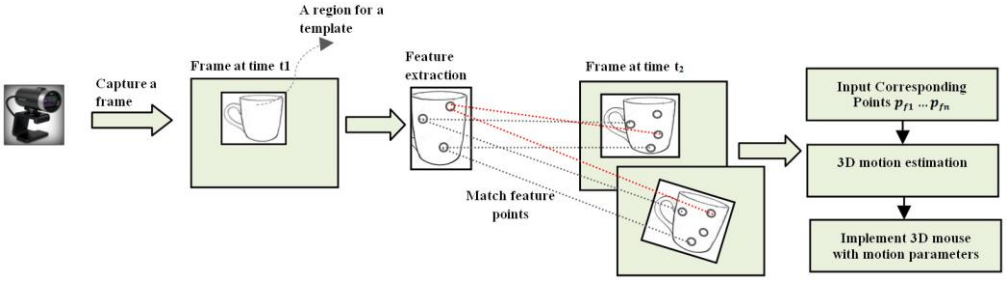

Figure 3. Process flow of the proposed method

For creation of the real dataset, we firstly captured video sequences for a moving object with 640×360 pixel resolution while changing object position in the front of the static camera along each axis and rotating around each axis across field of view of it. Note that object position was arbitrarily fixed before changing it. Distance of the moving object was varied 300 mm closer to 800 mm away from the camera in direction of z axis. Translation of the object was varied up to 200 mm in x and y direction. Rotation of the object around z axis was varied up to 90 degrees, and rotation of the object around x and y axis was varied up to 20 degrees. We used thousands of images of the different textured objects to check estimation accuracy of the motion parameters. Object templates used in experiments and their descriptions are illustrated in Table 1.

TABLE I. DATASET USED IN EXPERIMENT

Dataset for a real scene					
Template					
No. of points	115	104	78	151	159
Size	150×120				
Dataset for a simulated scene					
3D object	polygon	cube	pyramid	polygon	cube
Edge length	15	15	15	13	13
No. of Points	15	25	25	13	25

For the simulated dataset, we created different 3D objects such as polygon, pyramid and cube with different edge lengths, and manually measured up to 25 feature points with 2-3 pixels noise on the 3D objects. The simulated sequences for a moving 3D object were created by perspective projection with choosing largest focal length that keeps the object in field of view throughout sequences. Descriptions of simulated dataset are summarized in Table1. Then we translated the 3D object by up to 15 units along x, y and z axis, and rotated it by up to 30 degrees around x and y axis, and up to 90 degrees around z axis in Euclidean space. The simulated dataset consists of thousands of images.

To simplify notation of experiment results, we named estimation methods based on decomposition of essential matrix as CV_EM, decomposition of homography matrix as CV_H, relative orientation as PM_RO and homography based relative orientation as PM_H.

Firstly, we estimated motion parameters for real dataset. We checked the accuracy of the estimated rotation parameters around each axis for all datasets with comparing true (known) rotation parameters by analyzing their root mean square errors (RMSEs), absolute mean errors (MEs), maximum errors, and minimum errors produced from the proposed four methods as summarized in Table 2. For reference value, we manually measured corresponding features for every object in the image sequences, and their precise 3D motion was estimated.

TABLE II. COMPARISON OF ERROR ANALYSIS FOR REAL SCENE

Comparison of Maximum Error /degrees/				
Method	PM_H	CV_H	CV_EM	PM_RO
ω $(1^\circ \sim 20^\circ)$	1.867	1.993	1.997	1.863
φ $(1^\circ \sim 20^\circ)$	1.596	1.731	1.723	1.711
κ $(1^\circ \sim 90^\circ$	0.892	0.757	1.728	1.721
Comparison of Minimum Error /degrees/				
ω $(1^\circ \sim 20^\circ)$	0.00014	0.000	0.013	0.0009
φ $(1^\circ \sim 20^\circ)$	0.00045	0.001	0.005	0.0004
κ $(1^\circ \sim 90^\circ$	0.0051	0.00	0.0003	0.0002
Comparison of Mean Error /degrees/				
ω $(1^\circ \sim 20^\circ)$	0.538	0.564	1.096	0.578
φ $(1^\circ \sim 20^\circ)$	0.346	0.403	0.871	0.620
κ $(1^\circ \sim 90^\circ$	0.391	0.312	0.677	0.408
Comparison of RMS Error /degrees/				
ω $(1^\circ \sim 20^\circ)$	0.685	0.717	1.199	0.718
φ $(1^\circ \sim 20^\circ)$	0.346	0.403	0.871	0.620
κ $(1^\circ \sim 90^\circ$	0.434	0.364	0.818	0.507

As we can see in Table 2 that RMSEs of the rotation results were small and accurate for each four approaches. Particularly, in cases of real image sequences, planar homography methods such as PM_H and CV_H produced more negligible and comparable errors among four estimations since planar pattern was dominating in test dataset. Among them motion parameters from the estimation method as PM_H were especially more accurate for image sequences of the moving planar object. When matched feature correspondences were noisy, CV_E was very sensitive for it.

Secondly, we checked accuracy of the motion parameters for simulated dataset. To compare performance of the four estimations for 3D motion we used the same measures as the real data case. The results of comparison are summarized in Table 3.

As we see in Table 3 that the four estimations produced small errors in cases of the simulated datasets. Specifically,

motion parameters from PM_RO were the most accurate than other three approaches.

Combining the two cases, we observed that the approach, PM_H was producing more accurate results for real image sequences, and the approach, PM_RO was producing more accurate results for 3D object in simulated sequences. This implies that all four method tested worked successfully when decent corresponding features were provided. It was anticipated that the photogrammetric method based on relative orientation produced most accurate results as this method estimates rotational and positional parameters directly. On the other hands, the results with real data are very interesting. Homography based methods outperformed other essential matrix or relative orientation based methods regardless of computer vision or photogrammetric approaches under noisy situation. In particular, the observation that homography based photogrammetric method worked better than relative orientation based one supports the motivation of linking techniques in developed in photogrammetry and computer vision.

TABLE III. COMPARISON OF ERROR ANALYSIS FOR SIMULATED DATASET

Comparison of Maximum Error /degrees/				
Method	PM_H	CV_H	CV_EM	PM_RO
$\boldsymbol{\omega(1^\circ \sim 30^\circ)}$	1.647	1.615	1.731	1.589
$\boldsymbol{\varphi(1^\circ \sim 30^\circ)}$	1.986	1.891	1.993	1.983
$\boldsymbol{\kappa(1^\circ \sim 90^\circ)}$	1.937	1.828	1.801	1.492
Comparison of Minimum Error /degrees/				
$\boldsymbol{\omega(1^\circ \sim 30^\circ)}$	0.015	0.009	0.013	0.003
$\boldsymbol{\varphi(1^\circ \sim 30^\circ)}$	0.005	0.01	0.025	0.012
$\boldsymbol{\kappa(1^\circ \sim 90^\circ)}$	0.014	0.016	0.007	0.013
Comparison of Mean Error /degrees/				
$\boldsymbol{\omega(1^\circ \sim 30^\circ)}$	0.549	0.498	0.762	0.461
$\boldsymbol{\varphi(1^\circ \sim 30^\circ)}$	0.648	0.647	0.71	0.629
$\boldsymbol{\kappa(1^\circ \sim 90^\circ)}$	0.684	0.556	0.605	0.519
Comparison of RMS Error /degrees/				
$\boldsymbol{\omega(1^\circ \sim 30^\circ)}$	0.68	0.632	0.933	0.581
$\boldsymbol{\varphi(1^\circ \sim 30^\circ)}$	0.818	0.801	0.878	0.798
$\boldsymbol{\kappa(1^\circ \sim 90^\circ)}$	0.811	0.645	0.736	0.603

To assess real-time performance, we measured processing time of SIFT feature extraction and motion estimations by including RANSAC based elimination for large number of feature points. Processing time of SIFT feature extraction was speeding up in 0.2 seconds. For four estimations, processing time was around 0.0001 seconds.

CONCLUSION

In order to improve the robustness of the proposed methods in photogrammetry and computer vision, we tracked planar and non-planar objects in experiment level. The solution equations in photogrammetry were formulated as a non-linear least squares problem to obtain unique solution, and the equations in computer vision were formulated as linear solution to obtain the number of possible solutions. We estimated motion parameters by using different RANSAC based methods for each estimation. The results of estimations in both fields were accurate in high variation of translation and rotation change under favorable correspondences. For noisy situation, methods based on homography produced smaller errors. Processing speed was close to real-time processing.

ACKNOWLEDGMENT

The work in this paper was supported by the National Research Foundation of Korea (NRF) grant by the Korea government (No. NRF-2016R1A2B4013017) and National University of Mongolia (No. P2017-2469).

REFERENCES

[1] J. Weng, T. Huang, and N. Ahuja, "Motion and structure from two perspective views: Algorithms, error analysis, and error estimation," IEEE Trans Pattern and Machine Intell, vol.11, pp.451–476, 1989.

[2] R. Hartley and A. Zisserman, "In Multiple view in computer vision," Cambridge University Press, 2000.

[3] G. Chesi, "Estimation of the camera pose from image point correspondences through the essential matrix and convex optimization," IEEE International Conference on Robotics and Automation, 2009.

[4] K. Fathian and N.R. Gans, "A new approach for solving the five-point relative pose problem for vision-based estimation and control," IEEE American Control Conference, 2014.

[5] G. Chesi and K. Hashimoto, "Camera pose estimation from less than eight points in visual servoing," IEEE International Conference on Robotics and Automation, 2004.

[6] E. Malis and M. Vargas, "Deeper understanding of the homography decomposition for vision-based control," Research Report 6303, INRIA, 2007.

[7] K. Kim, V. Lepetit, and W. Woo, "Scalable real-time planar targets tracking for digilog books," Vis Comput., vol.26 pp.1145–1154, 2010.

[8] H. Bazargani, O. Bilaniuk, and R. Laganie're, "A fast and robust homography scheme for real-time planar target detection," J Real-Time Image Proc., pp.1–20, 2015.

[9] Y. Mae, J. Choi, H. Takahashi, K. Ohara, T. Takubo, and T. Arai, "Interoperable vision component for object detection and 3D pose estimation for modularized robot control," Mechatronics, vol.21 pp.983–992, 2011.

[10] E. Marchand, H. Uchiyama, and F. Spindler, "Pose estimation for augmented reality: A hands-on survey," IEEE Trans. Vis. Comput. Graph., vol.22 pp.2633–2651, 2016.

[11] B. K. P. Horn, "Relative orientation," Int. J. Comput. Vision, vol.4 pp.59–78, 1990.

[12] T. Schenk, "In Digital Photogrammetry," page 428. Terra Science, Laurelville, OH, USA, 1999.

[13] P. Wolf, B. DeWitt, and B.E. Wilkinson, "In Elements of Photogrammetry with Applications in GIS," McGraw-Hill Science, New York, USA, 2014.

[14] J. Kim and T. Kim, "Comparison of computer vision and photogrammetric approaches for epipolar resampling of image sequence," Sensors, 2016.

[15] J.C. McGlone, E.M. Mikhail, and J. Bethel, "In Manual of Photogrammetry," American Society of Photogrammetry and Remote Sensing, Bethesda, MD, USA, 2004.

[16] D. Nister, "An efficient solution to the five points relative pose problem," IEEE Trans. Pattern And. Machine Intell., 2004.