Homography Based Egomotion Estimation with a Common Direction

Olivier Saurer, Pascal Vasseur, Rémi Boutteau, Cédric Demonceaux, Marc Pollefeys and Friedrich Fraundorfer

Abstract—In this paper, we explore the different minimal solutions for egomotion estimation of a camera based on homography knowing the gravity vector between calibrated images. These solutions depend on the prior knowledge about the reference plane used by the homography. We then demonstrate that the number of matched points can vary from two to three and that a direct closed-form solution or a Gröbner basis based solution can be derived according to this plane. Many experimental results on synthetic and real sequences in indoor and outdoor environments show the efficiency and the robustness of our approach compared to standard methods.

Index Terms—Computer vision, egomotion estimation, homography estimation, structure-from-motion.

1 Introduction

Nowadays, point-based methods to estimate the motion of a camera are well known. If the camera is uncalibrated, eight or seven points are needed to estimate the fundamental matrix between two consecutive views [1]. When the intrinsic parameters of the camera are known, five points are then enough to estimate the essential matrix [2]. To decrease the sensitivity of these methods, a robust framework such as Random Sample Consensus (RANSAC) is necessary. Thus, reducing the number of needed matched points between views is important in terms of computation efficiency and of robustness improvement. For example, as shown in Figure 1, for a probability of success of 0.99 and a rate of outliers equal to 0.5, the number of RANSAC trials is divided by eight, if five points are used instead of eight. In the case of a robust estimation based on eight points, 1177 trials are necessary whereas 145 are sufficient if only five points are required. Thus, finding a minimal solution for egomotion estimation is important for robust real time applications.

However, reducing the number of necessary points is only possible if some hypotheses or supplementary data are available. For example, if we know a common direction between the two views, three points can then be used to estimate the full essential matrix [3]. Extreme situations appear when a planar non-holonomic motion is supposed [4] or when the metric velocity of a single camera can be estimated knowing its attitude and its acceleration [5]. In these cases, only one point allows to estimate the motion. These initial hypotheses or additional knowledges can then deal with the pose of the camera or with the 3D structure of the scene. For example, if the 3D points belong to a single plane, the egomotion estimation is reduced to a homography computation between two views, that can be calculated using only four points [1]. In many scenes and many applications, the scene plane hypothesis seems suitable. Indeed, in many scenarios such as indoor or street corridors and more generally in man made environments, this assumption holds.

Thus, in this paper we investigate the cases where at least one plane is present in the scene and where we have some partial knowledge about the pose of the camera. We suppose that we are able to extract a common direction between consecutive views and we can have some information about the normal of the considered plane. Obtaining a common direction can be easily performed thanks to an IMU (Inertial Measurement Unit) associated with the camera, which is often the case in mobile devices or UAV (Unmanned Aerial Vehicle). The coupling with a camera is then very easy and can then be used for different computer vision tasks [6], [7], [8], [9], [10]. Without any external sensor, this common direction can also be directly extracted from the images thanks to vanishing points [11] or horizon detection [12].

In this work, assuming the roll and pitch angles of the camera as known, we propose to find a minimal closed-form solution for homography estimation in man made environments. We will derive different solutions depending on the prior knowledge about the 3D scene :

- If the extracted points lie on the ground plane, we will see that only two points are required to estimate

- *Olivier Saurer and Marc Pollefeys are with the Computer Vision and Geometry Group in the Department of Computer Science, ETH Zürich, Switzerland.*
  E-mail: saurero@inf.ethz.ch, marc.pollefeys@inf.ethz.ch
- *Pascal Vasseur is with LITIS, Université de Rouen, France.*
  E-mail: pascal.vasseur@univ-rouen.fr
- *Rémi Boutteau is with IRSEEM, ESIGELEC, Rouen, France.*
  E-mail: remi.boutteau@esigelec.fr
- *Cédric Demonceaux is with Le2i, UMR CNRS 6306 Université de Bourgogne, France.*
  E-mail: cedric.demonceaux@u-bourgogne.fr
- *Friedrich Fraundorfer is with the Institute for Computer Graphics and Vision, TU Graz, Austria.*
  E-mail: fraundorfer@icg.tugraz.at

Manuscript received February 16, 2016; revised ?? ??, 20??.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2545663, IEEE Transactions on Pattern Analysis and Machine Intelligence

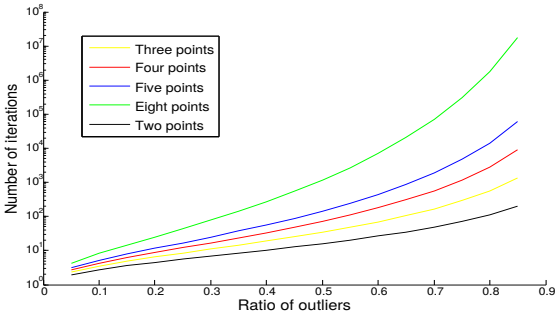

Fig. 1. Comparison of the RANSAC iteration number for 99% of success probability

the camera egomotion. In this case, the solution is unique and contrary to the other algorithms for essential matrix estimation, there is no supplementary verification for finding the good solutions among the different possibilities.

- If the considered points are on a vertical plane, we propose an efficient 2.5pt formulation in order to retrieve the motion of the camera and the normal of the plane related to the pose of the camera. This solution allows for an early reject of a pose hypothesis by including a consistency check on the three point correspondences.
- If the plane orientation is completely unknown, we develop a minimal solution using only three points instead of four points needed in the classical homography estimation.

All these methods will be evaluated on synthetic and real data and compared with different methods proposed in the literature.

The rest of the paper is organized as follows. In the second part, we describe the different existing methods in the literature which deal with minimal solution for egomotion estimation. In the next section, we explain how to reduce the number of points for estimating the homography between two views and derive the proposed solutions according to the prior knowledge. In the fourth section, we show the behaviour of our solutions on synthetic and real data and compare with other classical methods in a quantitative evaluation. Finally, we will conclude by providing some extents to this work.

2 Related Works

When the camera is not calibrated, at least 8 or 7 points are needed to recover the motion between views [1]. It's well known, that if the camera is calibrated, only 5 feature point correspondences are sufficient to estimate the relative camera pose. Reducing this number of points can be very interesting in order to reduce the computation time and to increase the robustness when using a robust estimator such as RANSAC. The reduction of the degree of freedom (DoF) number and consequently the number of matched points between images can be achieved by introducing some constraints on the camera motion (planar for example) or the feature points (on the same plane) or by using some additional information provided by other sensors such as IMU for instance.

For example, if all the 3D points lie on a plane, a minimum of 4 points is required to estimate the motion of the camera between two-views [1]. On the other hand, if the camera is embedded on a mobile robot which moves on a planar surface, only 2 points are required to recover the motion [13] and if in addition the mobile robot has non-holonomic constraints only one point is necessary [4]. Similarly, if the camera moves in a plane perpendicular to the gravity, 1 point correspondence is sufficient to recover the motion as shown by Troiani et al. [14].

The number of points needed to estimate the egomotion can be also reduced if some information about the relative rotation between two poses are available. This information can be given by vanishing points extraction in the images [15] or by taking into account extra information given by an additional sensor. Thus, Li et al. [16] show that in the case of an IMU associated to the camera, only 4 points are sufficient to estimate the relative motion even if the extrinsic calibration between the IMU and the camera is not known.

Similarly, some different algorithms have been recently proposed in order to estimate the relative pose between two cameras by knowing a common direction. It has been demonstrated that knowing roll and pitch angles of the camera at each frame, only three points are needed to recover the yaw angle and the translation of the camera motion up to scale [3], [17], [18]. In these approaches, only the formulation of the problem is different and consequently the way to solve it. All these works start with a simplified essential matrix in order to derive a polynomial equation system. For example, in [17], their parametrization leads to 12 solutions by using the Macaulay matrix method. The correct solution has then to be found among a set of possible solutions. The approach presented in [3] permits to obtain a 4^{th}-order polynomial equation and consequently leads to a more efficient solution. In [18], the authors propose a closed-form solution to this 4^{th}-order polynomial equation that allows a faster computation.

For a further reduction of necessary feature points, stronger hypotheses have to be added. If the complete rotation between the two views are known, only 2 degrees of freedom corresponding to the translation up-to-scale has to be estimated and consequently 2 points are sufficient to solve the problem [19], [20]. In this case, the authors compute the translation vector using the epipolar geometry given the rotation. Thus, these approaches allow to reduce the number of points but also imply the knowledge of the complete rotation between two views making the pose estimation very sensitive to IMU inaccuracy. More recently, Martinelli [21] proposes a closed-form solution for structure from motion knowing the gravity axis of the camera in a multiple view scheme. He shows that at least three feature points lying on a same plane and three consecutive views are required to estimate the motion. In the same way, the plane constraint has been used for reducing the complexity of the bundle adjustment (BA) in a visual simultaneous localization and mapping (SLAM) embedded on a micro-aerial vehicle (MAV) [22].

Most closely related papers to our approach are the works of [3], [18] in which they simplify the essential matrix

knowing the vertical of the cameras. In this work, to reduce the number of points, rather than deriving the epipolar constraint to compute the essential matrix, we propose to use the homography constraint between two views. Thus, we suppose that a significant plane exists in the scene and that the gravity direction is known. Let us note that recently, in [23] Troiani et al. have also proposed a method using 2 points on the ground plane with the knowledge of the vertical of the camera. However, they do not use the homography formalism and their method requires to know the distance between the two 3D points. In our method, this hypothesis is not necessary and we only assume that the points lie on a same plane. The Manhattan world assumption [24] has also recently successfully been used for multi-view stereo [25], the reconstruction of building interiors [26] and also for scene reconstruction from a single image only [27]. Our contribution differs from them, as we combine gravity measurements with the weak Manhattan world assumption. This paper is an extension of [28], [29] where we studied camera pose estimation based on homographies with a common vertical direction and a known or at least partially known plane normal. In [28] we proposed a homography based pose estimation algorithm that does not require any knowledge on the plane normal. In fact the algorithm provides the plane normal in addition to the camera pose.

3 Motion Estimation

Knowing the vertical direction in images will simplify the estimation of camera pose and camera motion, which are fundamental methods in 3D computer vision. It is then possible to align every camera coordinate system with the measured vertical direction such that the z-axis of the camera is parallel to the vertical direction and the x-y-plane of the camera is orthogonal to the vertical direction (illustrated in Fig. 2). In addition, this would mean that the x-y-plane of the camera is now parallel to the world's ground plane and the z-axis is parallel to vertical walls.

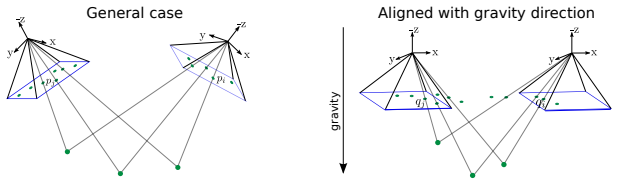

Fig. 2. Alignment of the camera with the gravity direction.

This alignment can just be done as a coordinate transform for motion estimation algorithms, but also be implemented as image warping such that feature extraction methods benefit from it. Relative motion between two such aligned cameras reduces to a 3-DOF motion, which consists of 1 remaining rotation and a 2-DOF translation vector (i.e., a 3D translation vector up to scale).

The algorithms for estimating the relative pose are derived from a homography formulation, where a plane is observed in two images. The homography is then decomposed into a relative rotation and translation between the two images. By incorporating the known vertical direction, the parametrization of the pose estimation problem is greatly reduced from 5-DOF to 3-DOF. This simplification leads to a closed-form 2pt and a 2.5pt algorithm to compute the homography. By relaxing the assumption of strictly vertical or horizontal structures and making use of the known gravity direction, the homography formulation results in a closed form solution requiring 3-points only.

In the following subsections we derive the 2pt algorithm for the known plane normal cases (ground and vertical plane), then we provide a derivation of the 2.5pt and 3pt algorithm for a known gravity direction with an unknown plane orientation.

3.1 2pt Relative Pose for Points on the Ground Plane

The general homographic relation for points belonging to a 3D plane and projected in two different views is defined as follows :

$$\mathbf{q}_j = \mathbf{H}\mathbf{q}_i, \tag{1}$$

with $\mathbf{q}_i = [x_i, y_i, w_i]^\top$ and $\mathbf{q}_j = [x_j, y_j, w_j]^\top$ the projective coordinates of the points between the views i and j. $\mathbf{H}$ is given by:

$$\mathbf{H} = \mathbf{R} - \frac{1}{d}\mathbf{t}\mathbf{n}^\top, \tag{2}$$

where $\mathbf{R}$ and $\mathbf{t}$ are respectively the rotation and the translation between views i and j and where d is the distance between the camera i and the 3D plane described by the normal $\mathbf{n}$.
In our case, we assume that the camera intrinsic parameters are known and that the points $\mathbf{q}_i$ and $\mathbf{q}_j$ are normalized. We also consider that the attitude of the cameras for the both views are known and that these attitude measurements have been used to align the camera coordinate system with the ground plane. In this way, only the yaw angle θ between the two views remains unknown. Therefore equation 2 can be expressed as:

$$\mathbf{H} \;=\; \mathbf{R}_z - \frac{1}{d}\mathbf{t}\mathbf{n}, \tag{3}$$

where $\mathbf{R}_z$ denotes the unknown rotation around the yaw angle (z-axis). Similarly, since we consider that the ground plane constitutes the visible 3D plane during the movement of the camera, we can note that $\mathbf{n} = [0, 0, 1]^\top$.
Consequently, equation 3 can be written as:

$$\mathbf{H} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} - \frac{\mathbf{t}}{d}\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^\top, \tag{4}$$

d being unknown, the translation can be known only up to scale. Consequently, the camera-plane distance d is set to 1 and absorbed by $\mathbf{t}$. We then obtain:

$$\mathbf{H} \;=\; \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^\top, \tag{5}$$

$$= \begin{bmatrix} \cos(\theta) & -\sin(\theta) & -t_x \\ \sin(\theta) & \cos(\theta) & -t_y \\ 0 & 0 & 1 - t_z \end{bmatrix}. \tag{6}$$

In a general manner, this homography can be parametrized as

$$\mathbf{H} = \begin{bmatrix} h_1 & -h_2 & h_3 \\ h_2 & h_1 & h_4 \\ 0 & 0 & h_5 \end{bmatrix}. \quad (7)$$

The problem consists of solving for the five entries of the homography $\mathbf{H}$. We consider the following relation:

$$\mathbf{q}_j \times \mathbf{H}\mathbf{q}_i = \mathbf{0}, \quad (8)$$

where $\times$ denotes the cross product. By rewriting the equation, we obtain:

$$\begin{bmatrix} x_j \\ y_j \\ w_j \end{bmatrix} \times \begin{bmatrix} h_1 & -h_2 & h_3 \\ h_2 & h_1 & h_4 \\ 0 & 0 & h_5 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ w_i \end{bmatrix} = \mathbf{0}. \quad (9)$$

This gives us three equations, where two of them are linearly independent. We expand the above equation and consider only the first two linearly independent equations, which results in:

$$\begin{bmatrix} -w_j y_i h_1 - w_j x_i h_2 - w_i w_j h_4 + w_i y_j h_5 \\ w_j x_i h_1 - w_j y_i h_2 + w_i w_j h_3 - w_i x_j h_5 \end{bmatrix} = \mathbf{0}. \quad (10)$$

The equation system can be re-written into:

$$\begin{bmatrix} -w_j y_i & -w_j x_i & 0 & -w_i w_j & w_i y_j \\ w_j x_i & -w_j y_i & w_i w_j & 0 & -w_i x_j \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{bmatrix} = \mathbf{0}. \quad (11)$$

The above equation represents a system of equations of the form $\mathbf{A}\mathbf{h} = 0$. It is important to note that $\mathbf{A}$ has rank 4. Since each point correspondence gives rise to two independent equations, we require two point correspondences to solve for $\mathbf{h}$ up to one unknown scale factor. The singular vector of $\mathbf{A}$, which has the smallest singular value spans a one dimensional (up to scale) solution space. We chose the solution $\mathbf{h}$ such that $||\mathbf{h}|| = 1$. Then, to obtain valid rotation parameters we enforce the trigonometric constraint $h_1^2 + h_2^2 = 1$ on $\mathbf{h}$, by dividing the solution vector by $\pm\sqrt{h_1^2 + h_2^2}$. The camera motion parameters, can directly be derived from the homography:

$$\mathbf{t} = \begin{bmatrix} -h_3, & -h_4, & 1 - h_5 \end{bmatrix}^\top, \quad (12)$$

$$\mathbf{R} = \begin{bmatrix} h_1 & -h_2 & 0 \\ h_2 & h_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (13)$$

Due to the sign ambiguity in $\pm\sqrt{h_1^2 + h_2^2}$ we obtain two possible solutions for $\mathbf{R}$ and $\mathbf{t}$. An alternative solution is proposed in Appendix A which uses an inhomogeneous system of equations to solve for the unknown camera pose.

3.2 2pt Relative Pose for a Known Vertical Plane Normal

The following algorithm is able to compute the relative pose given 2 point correspondences and the normal of the plane on which the points reside. The derivation will be carried out for a vertical plane but works similar for planes parametrized around other axis.

The homography for a vertical plane can be written as:

$$\mathbf{H} = \mathbf{R}_z - [t_x, t_y, t_z]^\top [n_x, n_y, 0], \quad (14)$$

where $\mathbf{R}_z$ denotes the rotation matrix around the z-axis. Expanding the expression in (14) we obtain:

$$\mathbf{H} = \begin{bmatrix} \cos(\theta) - n_x t_x & -\sin(\theta) - n_y t_x & 0 \\ \sin(\theta) - n_x t_y & \cos(\theta) - n_y t_y & 0 \\ -n_x t_z & -n_y t_z & 1 \end{bmatrix}, \quad (15)$$

$$= \begin{bmatrix} h_1 & h_2 & 0 \\ h_3 & h_4 & 0 \\ h_5 & \frac{n_y}{n_x} h_5 & 1 \end{bmatrix}. \quad (16)$$

This leaves 5 entries in $\mathbf{H}$ to be estimated. Each point correspondence gives 2 inhomogeneous linearly independent equations of the form $\mathbf{A}\mathbf{h} = \mathbf{b}$. Using equation 8 we obtain:

$$\begin{bmatrix} -d - h_3 a - h_4 b + h_5 x_i y_j + h_5 y_i c \\ -e + h_1 a + h_2 b - h_5 x_i x_j - h_5 x_j c \end{bmatrix} = \mathbf{0}, \quad (17)$$

$$\begin{bmatrix} 0 & 0 & -a & -b & x_i y_j + y_i c \\ a & b & 0 & 0 & -x_i x_j - x_j c \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{bmatrix} = \begin{bmatrix} d \\ e \end{bmatrix}, \quad (18)$$

with:

$$a = w_j x_i,\ b = w_j y_i,\ c = y_j \frac{n_y}{n_x},\ d = -w_i y_j,\ e = w_i x_j.$$

Using 2 point correspondences, this gives 4 equations which is a deficient-rank system. The solution is $\mathbf{h} = \mathbf{V}\mathbf{y} + \lambda\mathbf{v}$ (see [1]) where $svd(\mathbf{A}) = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ and $\mathbf{v}$ is the last column vector of $\mathbf{V}$. The vector $\mathbf{y}$ is computed by $\mathbf{y}_i = \mathbf{b}'_i / d_i$ where d_i is the i^{th} diagonal entry of $\mathbf{D}$ and $\mathbf{b}' = \mathbf{U}^\top \mathbf{b}$.

This leaves the unknown scalar λ which can be computed from the additional trigonometric constraint $(\cos^2(\theta) + \sin^2\theta) - 1 = 0)$.

The trigonometric constraint can be fully expressed in terms of the variables h_1, h_2, h_3, h_4.

$$\cos(\theta)^2 + \sin(\theta)^2 - 1 = 0, \quad (19)$$

$$(h_1 + n_x t_x)^2 + (-h_2 - n_y t_x)^2 - 1 = 0, \quad (20)$$

with:

$$t_x = n_x(h_4 - h_1) - n_y(h_2 + h_3). \quad (21)$$

Substituting symbolically the entries of $\mathbf{h} = \mathbf{V}\mathbf{y} + \lambda\mathbf{v}$ into Eq. 20 results in a quadratic equation in the remaining unknown λ (the expanded equation is not shown due to its excessive length). Solving for the variable λ gives two solutions for the parameters h_1, h_2, h_3, h_4, h_5.

Once the homography $\mathbf{H}$ is estimated it can be decomposed into relative rotation and relative translation parameters. We back-substitute the entries of $\mathbf{H}$ from (16), that is h_1, h_2, h_3, h_4 and h_5 into (15). Knowing n_x and n_y, the translation parameters can directly be computed using the following relations:

$$t_z = \frac{-h_5}{n_x}, \quad (22)$$
$$t_x = n_x(h_4 - h_1) - n_y(h_2 + h_3), \quad (23)$$
$$t_y = n_y(h_1 - h_4) - n_x(h_2 + h_3). \quad (24)$$

And the rotation parameter is then obtained through:

$$\cos(\theta) = h_1 + n_x t_x. \quad (25)$$

3.3 2.5pt Relative Pose with Unknown Vertical Plane Normal

The 2.5pt algorithm is an extension of the 2pt algorithm described in section 3.2. The homography is parametrized as in (14). However, when the plane normal $\mathbf{n}$ is not known it is not possible to make use of the same linear constraint, thus all the 6 parameters of $\mathbf{H}$ have to be estimated. To do so, one more equation is required which can be taken from a third point. Thus the constraint equations of 2 points and 1 of the equations from a third point are stacked into an equation system of the form $\mathbf{Ah} = \mathbf{b}$. For one point correspondence two equations can be derived as follows. First the homography is defined as:

$$\mathbf{H} = \begin{bmatrix} \cos(\theta) - n_x t_x & -\sin(\theta) - n_y t_x & 0 \\ \sin(\theta) - n_x t_y & \cos(\theta) - n_y t_y & 0 \\ -n_x t_z & -n_y t_z & 1 \end{bmatrix}, \quad (26)$$
$$= \begin{bmatrix} h_1 & h_2 & 0 \\ h_3 & h_4 & 0 \\ h_5 & h_6 & 1 \end{bmatrix}. \quad (27)$$

Computing $\mathbf{q}_j \times \mathbf{H}\mathbf{q}_i$ leads to:

$$\begin{bmatrix} -c - h_3 a - h_4 b + h_5 x_i y_j + h_6 y_i y_j \\ -d + h_1 a + h_2 b - h_5 x_i x_j - h_6 x_j y_i \end{bmatrix} = \mathbf{0}, \quad (28)$$

$$\begin{bmatrix} 0 & 0 & -a & -b & x_i y_j & y_i y_j \\ a & b & 0 & 0 & -x_i x_j & -x_j y_i \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \end{bmatrix} = \begin{bmatrix} c \\ d \end{bmatrix}, \quad (29)$$

with:

$$a = w_j x_i,\ b = w_j y_i,\ c = -w_i y_j,\ d = w_i x_j.$$

As in section 3.2 the solution to this system is of the form $\mathbf{h} = \mathbf{V}\mathbf{y} + \lambda\mathbf{v}$. The unknown scalar λ can be computed utilizing the trigonometric constraint and a constraint on the normal vector:

$$\cos^2(\theta) + \sin^2(\theta) - 1 = 0, \quad (30)$$
$$n_x^2 + n_y^2 = 1. \quad (31)$$

Starting from equation 30 the constraint can be derived by substituting $cos(\theta)$ and $sin(\theta)$ with expressions in h_1 and h_2. In a next step t_x is substituted with equation 21. The relation $n_x^2 + n_y^2 = 1$ can be used to cancel out many terms in the equation and one obtains:

$$h_1^2 n_y^2 + h_2^2 n_x^2 + h_3^2 n_y^2 + h_4^2 n_x^2 - 2(h_1 h_2 + h_3 h_4) n_x n_y - 1 = 0. \quad (32)$$

Using (34-36) the equation can be rewritten in terms of h_1, h_2, h_3, h_4, h_5:

$$h_1^2 h_6^2 - 2h_1 h_2 h_5 h_6 + h_2^2 h_5^2 + h_3^2 h_6^2 - 2h_3 h_4 h_5 h_6 + h_4^2 h_5^2 - h_5^2 - h_6^2 = 0. \quad (33)$$

Substituting symbolically the entries of $\mathbf{h} = \mathbf{V}\mathbf{y} + \lambda\mathbf{v}$ into equation 33 results in a 4^{th} order polynomial in the remaining unknown λ (the expanded equation is not shown due to its excessive length). Root solving for the variable λ gives 4 solutions for the parameter sets h_1, h_2, h_3, h_4, h_5.

The decomposition of the homography into translation and rotation parameters of the relative motion follows the same steps as the one in section 3.2. However, it differs as the normals n_x and n_y are not given and need to be computed in the process. We again back-substitute the entries of $\mathbf{H}$ from (27) into (26). First we compute t_z using the relation $n_x^2 + n_y^2 = 1$,

$$t_z = \pm\sqrt{h_5^2 + h_6^2}. \quad (34)$$

This gives two solutions for t_z which differ in the sign and which leads to two further sets of derived solution sets. Now the unknown normals can be computed.

$$n_x = \frac{-h_5}{t_z}, \quad (35)$$
$$n_y = \frac{-h_6}{t_z}. \quad (36)$$

After this the procedure of section 3.2 can be followed again to compute the remaining parameters with the following equations, however, using both solutions for t_z, n_x and n_y,

$$t_x = n_x(h_4 - h_1) - n_y(h_2 + h_3), \quad (37)$$
$$t_y = n_y(h_1 - h_4) - n_x(h_2 + h_3). \quad (38)$$

The angle θ can be computed from the relation

$$\cos(\theta) = h_1 + n_x t_x. \quad (39)$$

An interesting fact in this case is, that only one of the two available equations from the third point is used. Although for the RANSAC loop it is still necessary to sample 3 points for this method, it is now possible to do a consistency check on the third point correspondence. To be an outlier free homography hypothesis the one remaining equation has also to be fulfilled. This can easily be tested and if it is not fulfilled the hypothesis is prematurely rejected. This gives a computational advantage over the standard 3pt essential matrix method [3], because inconsistent samples can be detected without testing on all the other point correspondences.

3.4 3pt Relative Pose using the Homography Constraint

In this section we discuss a 3pt formulation of the camera pose estimation with a known vertical direction. It differs from the algorithms in the previous section as it does not need the presence of scene planes. A 3pt algorithm has already been presented by [3] but using an essential matrix formulation. With this 3pt algorithm we propose an alternative to the previous essential matrix algorithm but based on a homography formulation. We start from (14), instead of assuming the plane to be parallel to the gravity vector we don't make any assumption on the plane orientation and therefore use 3 parameters n_x, n_y, n_z, for the fully unknown plane normal, which leads to:

$$\mathbf{H} = \mathbf{R}_z - [t_x, t_y, t_z]^\top [n_x, n_y, n_z]. \tag{40}$$

The camera-plane distance is absorbed by $\mathbf{t}$ the same way as in the previous sections.

The homography matrix then consists of the following entries:

$$\mathbf{H} = \begin{bmatrix} \cos(\theta) - t_x n_x & -\sin(\theta) - t_x n_y & -t_x n_z \\ \sin(\theta) - t_y n_x & \cos(\theta) - t_y n_y & -t_y n_z \\ -t_z n_x & -t_z n_y & 1 - t_z n_z \end{bmatrix}. \tag{41}$$

The unknowns we are seeking for are the motion parameters $\cos(\theta), \sin(\theta), t_x, t_y, t_z$ and the normal $[n_x, n_y, n_z]$ of the plane spanned by the 3 point correspondences. Recall that the standard 3pt essential matrix algorithm only solves for the camera motion, while the 3pt homography algorithm provides the camera motion and a plane normal with the same number of correspondences. To solve for the unknowns we setup an equation system of the form: $\mathbf{q}_j \times \mathbf{H}\mathbf{q}_i = 0$ and expand the relations to obtain the following two polynomial equations:

$$\begin{aligned} a t_y - b t_z - w_j x_i \sin(\theta) - w_j y_i \cos(\theta) + y_j w_i &= 0, \quad (42) \\ -a t_x + c t_z + w_j x_i \cos(\theta) - w_j y_i \sin(\theta) - x_j w_i &= 0, \quad (43) \end{aligned}$$

where:

$$\begin{aligned} a &= w_j x_i n_x + w_j y_i n_y + w_j n_z w_i, \\ b &= y_j w_i n_z + y_j n_x x_i + y_j n_y y_i, \\ c &= x_j n_x x_i + x_j w_i n_z + x_j n_y y_i. \end{aligned} \tag{44}$$

The third equation obtained from $\mathbf{q}_j \times \mathbf{H}\mathbf{q}_i = 0$ is omitted since it is a linear combination of the two other equations. Therefore each point correspondence gives 2 linearly independent equations and there are two additional quadratic constraints, the trigonometric constraint and the unit length of the normal vector that can be utilized:

$$\begin{aligned} \sin^2(\theta) + \cos^2(\theta) &= 1, \quad (45) \\ n_x^2 + n_y^2 + n_z^2 &= 1. \quad (46) \end{aligned}$$

The total number of unknowns is 8 and the two quadratic constraints together with the equations from 3 point correspondences give a total of 8 polynomial equations in the unknowns. An established way to find an algebraic solution to such a polynomial equation system is by using the Gröbner basis technique [30]. By computing the Gröbner basis a univariate polynomial can be found which allows to find the value of the unknown variable by root solving. The remaining variables can then be computed by back-substitution. To solve our problem we use the automatic Gröbner basis solver by Kukelova et al. [31], which can be downloaded at the authors webpage. The software automatically generates Matlab-Code that computes a solution to the given polynomial equation system (in our case the above specified 8 equations). The produced Matlab-Code consists of 299 lines and thus cannot be given here. The analysis of the Gröbner basis solutions shows, that the final univariate polynomial has degree 8, which means that there are up to 8 real solutions to our problem.

TABLE 1
Comparison of the degenerate conditions (yes means degenerate) for the standard 3pt method, the proposed 3pt homography method, the 2pt methods and the 2.5pt method.

	3pt-essential	3pt-hom	2pt	2.5pt
collinear points	no	yes	no	no
collinear points parallel to translation direction	yes	yes	no	no
points coplanar to translation vector	yes	yes	no	no

3.5 Degenerate Configurations

In this section we discuss the degenerate conditions for the proposed algorithms. In previous works [3], [18], [17] the degenerate conditions for the standard 3pt method for essential matrix estimation have been investigated in detail. In these papers multiple degenerate conditions are identified. It is also pointed out that a collinear configuration of 3D points is in general not a degenerate condition for the 3pt method, while it is one for the 5pt method. Degenerate conditions for the standard 3pt algorithm however are collinear points that are parallel to the translation direction and points that are coplanar to the translation vector. We investigated if these scenarios also pose degenerate conditions for our proposed algorithms, the 2pt, 2.5pt and 3pt homography method by conducting experiments with synthetic data. Degenerate cases could be identified by a rank loss of the equation system matrix or for the Gröbner basis case as a rank loss of the action matrix. For the 3pt homography case this revealed that the proposed method shares the degenerate conditions of the standard 3pt method but in addition also has a degenerate condition for the case of collinear points. This is understandable as the 3pt homography method also solves for the plane normal which then has an undefined degree of freedom around the axis of the collinear points. For the 2pt (both 2pt methods share the same properties) and 2.5pt algorithm these special cases however, do not pose degenerate conditions. More information in case of knowledge or partial knowledge of plane parameters allows to avoid degeneracy in the cases critical for the more general 3pt methods. The results of the comparison are summarized in Table 1.

4 EXPERIMENTS

4.1 Synthetic Evaluation

To evaluate the algorithms on synthetic data we chose the following setup. The average distance of the scene to the first camera center is set to 1. The scene consists of two planes, one ground plane and one vertical plane which is parallel to the image plane of the first camera. Both planes consist of 200 randomly sampled points. The baseline between two cameras is set to be 0.2, i.e., 20% of the average scene distance, and the focal length is set to 1000 pixels, with a field of view of 45 degrees.
Each algorithm is evaluated under varying image noise and increasing IMU noise. Each of the two setups is evaluated under a forward, into the scene (along the z-axis) and a sideways (along the x-axis) translation of the second camera. In addition the second camera is rotated around each axis.

To evaluate the robustness of the algorithms we compare the relative translation and rotation separately. The error measure compares the angle difference between the true rotation and the estimated rotation. Since the translation is only known up to scale, we compare the angle between the true- and estimated translation. The errors are computed as follows:

- Angle difference in $\mathbf{R}$:
  $\xi_R = \arccos((\mathrm{Tr}(\mathbf{R}\dot{\mathbf{R}}^\top) - 1)/2)$
- Direction difference in $\mathbf{t}$:
  $\xi_t = \arccos((\mathbf{t}^\top\dot{\mathbf{t}})/(\|\mathbf{t}\|\|\dot{\mathbf{t}}\|))$

Where $\mathbf{R}$, $\mathbf{t}$ denote the ground-truth transformation and $\dot{\mathbf{R}}$, $\dot{\mathbf{t}}$ are the corresponding estimated transformations.

Each data point in the plots represents the 5-quantile[1] (Quintiles) of 1000 measurements.

4.1.1 Relative Pose

Fig. 3 and Fig. 4 compare the 2-point algorithm to the general 5pt-essential matrix [2], 4pt-homography [1] and 3pt-essential matrix [3] algorithms. Notice, in these experiments the camera poses were computed from points randomly drawn from the ground plane. Since camera poses estimated from coplanar points do not provide a unique solution for the 5pt, 4pt and 3pt-essential matrix algorithm we evaluate each hypothesis with all points coming from both planes. The solution providing the most inliers is chosen to be the correct one. This evaluation is used in all our synthetic experiments. Similarly Fig. 5 and Fig. 6 show a comparison of the 2.5pt algorithm with the general 5pt, the 4pt and the 3pt-essential matrix algorithms. Here the camera poses are computed from points randomly sampled from the vertical plane only.
The evaluation shows that knowing the vertical direction and exploiting the planarity of the scene improves motion estimation. The 2pt and 2.5pt algorithms outperform the 5pt and 4pt algorithm, in terms of accuracy. Under perfect IMU measurements the algorithms are robust to image noise and perform significantly better than the 5pt and 4pt algorithm. With increasing IMU noise their performance are still comparable to the 5pt algorithm and superior to the 4pt algorithm.

1. The k-quantile represents the boundary value of the k^{th} interval when dividing ordered data into k regular intervals. For $k = 2$, the 2-quantile represents the median value.

4.1.2 3pt Homography

Fig. 7 and Fig. 8 compare the 3pt-homography based algorithm to the general 5pt [2] and the 3pt-essential matrix algorithms [3]. The evaluation shows that the proposed method outperforms the 5pt algorithm, in terms of accuracy. Under perfect IMU measurements the algorithm is robust to image noise and performs significantly better than the 5pt algorithm and equally good as the 3pt-essential matrix algorithm. With increasing IMU noise the performance of the 3pt-essential matrix and 3pt-homography algorithms are still comparable to the 5pt algorithm.

4.1.3 Timings

We evaluate the run-time of all algorithms on an Intel i7-2600K 3.4GHz using Matlab. To provide a fair comparison all algorithms were implemented in Matlab. No mex files were used, except for the reduced row echelon function *rref*, which is required by the 3pt-essential and 3pt-homography algorithms. All timings were averaged over 1000 runs. Table 2 summarizes the run-times for each of the six algorithms. The high run time of the 3pt-homography algorithm is due to the complexity of the Gröbner basis solution, which has to perform Gauss-Jordan elimination on the 443x451 elimination matrix.

For one RANSAC iteration the timings can vary drastically between algorithms. This is due to the different solution spaces the algorithms provide. To have the same error measure for all algorithms, we choose to use the re-projection error to select the correct camera poses among a set of possible poses. For instance the 2pt algorithm provides one unique camera pose, while the 5pt algorithm can provide up to 10 different essential matrices. In addition for each essential matrix 4 possible camera poses exist and need to be verified to find the correct pose, which can result in a total of 40 possible camera poses. While the homography formulations directly provide sets of camera poses. Even though the hypothesis estimation of the 3pt-homography algorithm has a larger constant time complexity, compared to its essential matrix counter part, one RANSAC iteration is cheaper, since fewer potential poses need to be evaluated. The table clearly shows that the computation time is dominated by the hypothesis selection (re-projection error computation) and not by the solver. In all experiments we used a set of 200 point correspondences.

TABLE 2
Run-time comparison of different pose estimation algorithms. The second column provides timings for estimating the hypothesis. The third column provides timings for one RANSAC iterations, which includes the selection of the right solution from a set of hypothesis. The last column shows the average number of real solutions (camera poses) provided by the respective algorithm. See text for more details.

Method	Hypothesis Estimation(ms)	RANSAC 1 Iteration(ms)	Avg. # Solutions
2pt	0.09	8.31	2
2.5pt	0.22	33.45	8
3pt-homography	27.28	55.17	6.85
3pt-essential	0.49	25.02	6.18
4pt-homography	0.18	8.65	2
5pt-essential	0.42	64.33	16.02

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2545663, IEEE Transactions on Pattern Analysis and Machine Intelligence

4.2 Real Data Experiments

In the following section we evaluate the proposed algorithms on both an indoor and outdoor environment.

4.2.1 Error Measure

In order to compare the estimated camera poses to the ground-truth, we used the relative pose error (RPE) measure as proposed by Sturm [32]. The RPE compares the local accuracy of the trajectory over a fixed time interval Δ, that corresponds to the drift of the trajectory. The RPE at time step i can be defined as:

$$\mathbf{E}_i = (\mathbf{Q}_i^{-1}\mathbf{Q}_{i+\Delta})^{-1}(\mathbf{P}_i^{-1}\mathbf{P}_{i+\Delta}), \tag{47}$$

where $\mathbf{Q}_i, \mathbf{P}_i \in SE(3)$ represent the ground truth and estimated poses respectively. $\mathbf{E}_i$ then represents the relative error. For a sequence of n camera poses, $m = n - \Delta$ individual relative pose errors are then estimated. From these errors, we propose to compute the root mean squared error (RMSE) over all time indices of the translational component as

$$RMSE(\mathbf{E}_{1:n}, \Delta) = \sqrt{\frac{1}{m}\sum_{i=1}^{m} \|trans(E_i)\|^2}, \tag{48}$$

where $trans(\mathbf{E}_i)$ refers to the translational components of the relative pose error $\mathbf{E}_i$.

4.2.2 Vicon Dataset

In order to have a practical evaluation of the 2pt, 2.5pt and 3pt algorithms, several real datasets have been collected with reliable ground-truth, see Fig. 9. The ground-truth data has been obtained by conducting the experiments in a room equipped with a Vicon motion capture system made of 22 cameras. We used the Vicon data as inertial measures and scale factor in the different experiments. The sequences have been acquired with a perspective camera mounted either on teleoperated Segway mobile robot (Fig. 9) or with a handheld system in order to have planar and 3D trajectories. In both cases the cameras are synchronized with the Vicon system. The image resolution used is 1624×1234 pixels. The length of these trajectories is between 20 and 50 meters and the number of images is between 150 and 350 per sequence. Robot motion speed is about 1m/s. Two different sets have been acquired, one set showing the ground plane dominantly and another set showing the walls dominantly.

We perform a comparison of the 2pt, 2.5pt and 3pt-homography with the 5pt algorithm in order to show the efficiency of the proposed methods. First, we use [33] to extract and match SIFT [34] features. The same matched feature point sets are used for the different algorithms and form the input to RANSAC [35] in order to select the inliers. For RANSAC we use a fixed number of 100 iterations, in all our experiments.

Figure 10 shows the evaluation of the 2pt ground plane algorithm. The trajectories obtained with 2pt (red curve) and 5pt (black curve) are compared with the ground-truth (blue curve) from Vicon. In all these experiments, even if both approaches propose trajectories globally with a similar shape than the ground-truth, we can note that the 2pt algorithm provides better results than the 5pt method. In the case of planar trajectories, that is sequence I, II, IV in Fig. 10, it is worth noting that the 2pt algorithm has a very low drift in the vertical axis while the 5pt accumulates significant error. Over the six sequences, the mean angular error in translation is equal to 0.1883 radians for the 2pt and 0.3380 for the 5pt.

TABLE 3
Root Mean Squared Error overview.

Sequence	2pt (mm)	5pt (mm)
Ground Sequence I	8.94	48.55
Ground Sequence II	9.28	56.66
Ground Sequence III	18.46	65.39
Ground Sequence IV	14.25	93.30
Ground Sequence V	25.61	34.46
Ground Sequence VI	39.75	67.45

TABLE 4
Root Mean Squared Error overview for the *Wall Sequence*.

Method	Wall Sequence I	Wall Sequence II
2.5pt	60.65	41.60
5pt-essential	24.97	27.84
3pt-essential	27.44	64.65
3pt-homography	27.26	65.96

The root mean squared error as defined in equation 48 is given for all 6 sequences in Table 3. The 2pt clearly outperforms the 5pt algorithm, providing a $1.69 \times -6.54\times$ lower error compared to the 5pt algorithm.

Figure 11 compares the different trajectories obtained from the 2.5pt (red curve), the 3pt-homography algorithm (green curve), the 3pt-essential matrix algorithm (magenta curve) and the 5pt (black curve), to the ground-truth (blue curve) obtained from the Vicon system.

In Table 4 we compare the RMS-error of the different algorithms. The 2.5pt algorithm shows similar performance compared to the standard 5pt algorithm and both 3pt algorithms, however having the advantage of a much simpler derivation.

This experiments also demonstrated that the assumptions taken for the 2pt algorithm (flat ground plane) and for the 2.5pt algorithm (vertical walls) are met in practical situations and can be used in real applications.

4.2.3 2pt Algorithm in a SFM Pipeline

In this final experiment we demonstrate the usage of the 2pt algorithm within an incremental SFM pipeline. The 2pt algorithm is used to replace the 5pt algorithm within the SFM pipeline. For this experiment the MAVMAP [36] SFM pipeline has been adapted to compare the 2pt algorithm to the 5pt algorithm. Two-view pose estimation is used when processing each new frame. To compute the relative pose between two consecutive frames we estimate the essential matrix in case of the 5pt algorithm and the homography for the 2pt algorithm. Afterwards full bundle adjustment is performed to compute precise camera poses and 3D points. The main goal of this experiment is to show that the 2pt can in practice replace standard algorithms (like the 5pt) for gaining a speed up but by maintaining the accuracy of the system.

For this experiment a UAV data set of a parking lot (denotated ParkingLot data set) is used. Images were captured by a gimbal mounted camera, such that the z-axis of the camera aligns with the gravity direction. The dataset was recorded at a native resolution of 24MP. The UAV was equipped with GPS and the GPS trajectory is utilized as ground truth for comparison. Figure 12 shows the results for this experiment. Figure 12(a) shows the output of the SFM system, resulting 3D point cloud (densified with SURE [37]), camera positions (red) and GPS positions (green). Figure 12(b) shows RPE plots for an experiment using the 5pt (black) algorithm and the 2pt (red) algorithm. Both algorithms lead to almost identical results. The value of the remaining RPE error is mainly due to the uncertainty of the GPS measurements and expected in this form. The resulting re-projection error after bundle adjustment is 0.249px for the 2pt case and 0.246px for the 5pt case, almost identical. To be clear, the reason for the identical re-projection error comes from bundle adjustment. This experiment demonstrates that the proposed 2pt algorithm can successfully replace the standard 5pt in a SFM system seamlessly but with the advantage of a gained speed-up.

5 Conclusion

In this paper we presented novel algorithms for relative pose estimation. The proposed methods differ from previous algorithms by utilizing a known common direction and assumptions about the environment. This makes it possible to derive algorithms that need less point correspondences for relative motion estimation as compared to state-of-the-art methods. This leads to improved RANSAC performance, which is a fundamental building block of any motion estimation or SFM system. In our paper we show through a variety of experiments with synthetic and real data the usability of the method. In particular, the results of our real world experiments clearly demonstrate that the assumptions about ground plane and vertical walls really hold in typical usage scenarios. Our algorithms have successfully been used for indoor robot navigation as well as for 3D reconstruction of aerial images taken from a UAV.

In addition to pure ego-motion estimation, there exists another obvious use case for our proposed algorithms, which has not been investigated in this work. Our methods not only provide the motion between views but are also able to detect the different scene planes of the environment. In an environment consisting of vertical walls and ground plane, the proposed 2pt methods could do plane detection more efficient and robust as compared to the otherwise used general 4pt homography method.

Appendix A
An Alternative Solution to the 2pt Homography Formulation

The general homographic relation for points belonging to a 3D plane and projected into two different views is defined as follows:

$$\mathbf{q}_j = \mathbf{H}\mathbf{q}_i, \tag{49}$$

with $\mathbf{q}_i = [x_i, y_i, w_i]^\top$ and $\mathbf{q}_j = [x_j, y_j, w_j]^\top$ the projective coordinates of the points between the views i and j. $\mathbf{H}$ is given by:

$$\mathbf{H} = \mathbf{R} - \frac{1}{d}\mathbf{t}\mathbf{n}^\top, \tag{50}$$

where $\mathbf{R}$ and $\mathbf{t}$ are respectively the rotation and the translation between view i and j and where d is the distance between the camera i and the 3D plane described by the normal $\mathbf{n}$.
In our case, we assume that the camera intrinsic parameters are known and that the points $\mathbf{q}_i$ and $\mathbf{q}_j$ are normalized. We also consider that the attitude of the cameras for the both views are known and that these attitude measurements have been used to align the camera coordinate system with the ground plane. In this way, only the yaw angle θ between the two views remains unknown. Similarly, since we consider that the ground plane constitutes the visible 3D plane during the movement of the cameras, we can note that $\mathbf{n} = [0, 0, 1]^\top$.
Consequently, equation(50) can be written as:

$$\mathbf{H} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} - \frac{\mathbf{t}}{d}\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^\top, \tag{51}$$

d being unknown, translation can be known only up to scale. Consequently, the camera-plane distance d is set to 1 and absorbed by $\mathbf{t}$. We then obtain:

$$\mathbf{H} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^\top \tag{52}$$

$$= \begin{bmatrix} \cos(\theta) & -\sin(\theta) & -t_x \\ \sin(\theta) & \cos(\theta) & -t_y \\ 0 & 0 & 1 - t_z \end{bmatrix}. \tag{53}$$

If $1 - t_z \neq 0$, then the homography can be scaled, by deviding $\mathbf{H}$ by $1 - t_z$, which results in:

$$\mathbf{H} = \begin{bmatrix} \frac{\cos(\theta)}{1-t_z} & -\frac{\sin(\theta)}{1-t_z} & -\frac{t_x}{1-t_z} \\ \frac{\sin(\theta)}{1-t_z} & \frac{\cos(\theta)}{1-t_z} & -\frac{t_y}{1-t_z} \\ 0 & 0 & 1 \end{bmatrix}. \tag{54}$$

In a general manner, this homography can be written as

$$\mathbf{H} = \begin{bmatrix} h_1 & -h_2 & h_3 \\ h_2 & h_1 & h_4 \\ 0 & 0 & 1 \end{bmatrix}. \tag{55}$$

The problem consists then in estimating this homography. In this way, we consider the following relation:

$$\mathbf{q}_j \times \mathbf{H}\mathbf{q}_i = \mathbf{0}, \tag{56}$$

where $\times$ denotes the cross product. In our case, we obtain:

$$\begin{bmatrix} x_j \\ y_j \\ w_j \end{bmatrix} \times \begin{bmatrix} h_1 & -h_2 & h_3 \\ h_2 & h_1 & h_4 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_i \\ y_i \\ w_i \end{bmatrix} = \mathbf{0}, \tag{57}$$

$$\begin{bmatrix} w_i y_j - w_j(h_2 x_i + h_1 y_i + h_4 w_i) \\ -w_i x_j + w_j(h_1 x_i + h_2 y_i + h_3 w_i) \\ x_j(h_2 x_i + h_1 y_i + h_4 w_i) - y_j(h_1 x_i + h_2 y_i + h_3 w_i) \end{bmatrix} = \mathbf{0}. \tag{58}$$

The third equation being a linear combination of the two others, we only consider the two first lines of the previous system and obtain:

$$\left[\begin{array}{c} w_i y_j - w_j h_2 x_i - w_j h_1 y_i - w_j h_4 w_i \\ -w_i x_j + w_j h_1 x_i - w_j h_2 y_i - w_j h_3 w_i) \end{array} \right] = \mathbf{0}, \quad (59)$$

which can be finally rewritten as

$$\left[\begin{array}{cccc} -w_j y_i & -w_j x_i & 0 & -w_j w_i \\ w_j x_i & -w_j y_i & w_j w_i & 0 \end{array} \right] \left[\begin{array}{c} h_1 \\ h_2 \\ h_3 \\ h_4 \end{array} \right] = \left[\begin{array}{c} -w_i y_j \\ w_i x_j \end{array} \right]. \quad (60)$$

We can observe that two points are sufficient in order to estimate the $h_l, l = \{1, 2, 3, 4\}$ elements and that the system can be easily extended to n points in order to obtain an over determined system. This possibility is then very interesting because it permits to apply a RANSAC algorithm in order to reject the outliers.

Once the $h_l, l = \{1, 2, 3, 4\}$ estimated, we are then able to evaluate the motion parameters, i.e., θ, t_x, t_y and t_z. By identification between (55) and (54), we know that $h_1 = \frac{\cos(\theta)}{1-t_z}$ and $h_2 = \frac{\sin(\theta)}{1-t_z}$. Then, if $h_1 \neq 0$, we have $\tan(\theta) = \frac{h_2}{h_1}$ and finally $\theta = \arctan(\frac{h_2}{h_1})$. Due to the periodicity of $\arctan$ two solutions arises, θ and $\theta + \pi$. If $h_1 = 0$, then $\theta = \pm\frac{\pi}{2}$.

Knowing the angle yaw θ between the two views and if $h_1 \neq 0$, we then have $t_z = 1 - \frac{\cos(\theta)}{h_1}$. If $h_1 = 0$, then $\theta = \pm\frac{\pi}{2}$ and we can use the relation $h_2 = \frac{\sin(\theta)}{1-t_z}$, to obtain $t_z = 1 \pm \frac{1}{h_2}$. Consequently, we can deduce the two remaining translation parameters $t_x = h_3(t_z - 1)$ and $t_y = h_4(t_z - 1)$.

ACKNOWLEDGMENTS

This work has been partially supported by Project ANR Blanc International DrAACaR-ANR-11-IS03-0003 and a Google Award. We also thank the anonymous reviewers for their useful discussions and constructive comments.

REFERENCES

[1] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[2] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 756–777, 2004.

[3] F. Fraundorfer, P. Tanskanen, and M. Pollefeys, "A minimal case solution to the calibrated relative pose problem for the case of two known orientation angles," in *ECCV (4)*, ser. Lecture Notes in Computer Science, K. Daniilidis, P. Maragos, and N. Paragios, Eds., vol. 6314. Springer, 2010, pp. 269–282.

[4] D. Scaramuzza, "1-point-ransac structure from motion for vehicle-mounted cameras by exploiting non-holonomic constraints," *International Journal of Computer Vision*, vol. 95, no. 1, pp. 74–85, 2011.

[5] L. Kneip, A. Martinelli, S. Weiss, D. Scaramuzza, and R. Siegwart, "Closed-form solution for absolute scale velocity determination combining inertial measurements and a single feature correspondence," in *IEEE International Conference on Robotics and Automation, ICRA 2011, Shanghai, China, 9-13 May 2011*, 2011, pp. 4546–4553.

[6] T. Viéville, E. Clergue, and P. D. S. Facao, "Computation of ego-motion and structure from visual and inertial sensors using the vertical cue," in *Computer Vision, 1993. Proceedings., Fourth International Conference on*. IEEE, 1993, pp. 591–598.

[7] P. Corke, "An inertial and visual sensing system for a small autonomous helicopter," *Journal of Robotic Systems*, vol. 21, no. 2, pp. 43–51, 2004.

[8] J. Lobo and J. Dias, "Vision and inertial sensor cooperation using gravity as a vertical reference," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1597–1608, 2003.

[9] S. Weiss and R. Siegwart, "Real-time metric state estimation for modular vision-inertial systems," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 4531–4537.

[10] J. Domke and Y. Aloimonos, "Integration of visual and inertial information for egomotion: a stochastic approach," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*. IEEE, 2006, pp. 2053–2059.

[11] J.-C. Bazin, H. Li, I. S. Kweon, C. Demonceaux, P. Vasseur, and K. Ikeuchi, "A branch-and-bound approach to correspondence and grouping problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 7, pp. 1565–1576, 2013.

[12] O. Oreifej, N. da Vitoria Lobo, and M. Shah, "Horizon constraint for unambiguous uav navigation in planar scenes," in *IEEE International Conference on Robotics and Automation, ICRA 2011, Shanghai, China, 9-13 May 2011*, 2011, pp. 1159–1165.

[13] D. Ortin and J. Montiel, "Indoor robot motion based on monocular images," *Robotica*, vol. 19, no. 3, pp. 331–342, 2001.

[14] C. Troiani, A. Martinelli, C. Laugier, and D. Scaramuzza, "1-point-based monocular motion estimation for computationally-limited micro aerial vehicles," in *Mobile Robots (ECMR), 2013 European Conference on*. IEEE, 2013, pp. 13–18.

[15] J.-C. Bazin, C. Demonceaux, P. Vasseur, and I. Kweon, "Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment," *The International Journal of Robotics Research*, vol. 31, no. 1, pp. 63–81, 2012.

[16] B. Li, L. Heng, G. H. Lee, and M. Pollefeys, "A 4-point algorithm for relative pose estimation of a calibrated camera with a known relative rotation angle," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS 2013, Tokyo, Japan*, 2013.

[17] M. Kalantari, A. Hashemi, F. Jung, and J.-P. Guédon, "A new solution to the relative orientation problem using only 3 points and the vertical direction," *Journal of Mathematical Imaging and Vision*, vol. 39, no. 3, pp. 259–268, 2011.

[18] O. Naroditsky, X. S. Zhou, J. H. Gallier, S. I. Roumeliotis, and K. Daniilidis, "Two efficient solutions for visual odometry using directional correspondence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 818–824, 2012.

[19] C. Troiani, A. Martinelli, C. Laugier, and D. Scaramuzza, "2-point-based outlier rejection for camera-imu systems with applications to micro aerial vehicles," in *IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, 2014.*, 2014.

[20] J.-C. Bazin, C. Demonceaux, P. Vasseur, and I. Kweon, "Motion estimation by decoupling rotation and translation in catadioptric vision," *Computer Vision and Image Understanding*, vol. 114, no. 2, pp. 254–273, 2010.

[21] A. Martinelli, "Closed-form solution of visual-inertial structure from motion," *International Journal of Computer Vision*, vol. 106, no. 2, pp. 138–152, 2014.

[22] G. H. Lee, F. Fraundorfer, and M. Pollefeys, "Mav visual slam with plane constraint," in *ICRA*. IEEE, 2011, pp. 3139–3144.

[23] C. Troiani, S. Al Zanati, and A. Martinelli, "A 3 Points Vision Based Approach for MAV Localization in GPS Denied Environments," in *6th European Conference on Mobile Robots*, Barcelona, Spain, 2013. [Online]. Available: http://hal.inria.fr/hal-00909613

[24] J. M. Coughlan and A. L. Yuille, "Manhattan world: Compass direction from a single image by bayesian inference," in *Internation Conference on Computer Vision*, 1999, pp. 941–947.

[25] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski, "Manhattan-world stereo," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. Miami, FL: IEEE, June 2009.

[26] ——, "Reconstructing building interiors from images." in *Proc. of the International Conference on Computer Vision (ICCV)*, 2009.

[27] D. C. Lee, M. Hebert, and T. Kanade, "Geometric reasoning for single image structure recovery," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[28] O. Saurer, F. Fraundorfer, and M. Pollefeys, "Homography based visual odometry with known vertical direction and weak manhattan world assumption," in *Vicomor Workshop at IROS*, 2012.

[29] O. Saurer, P. Vasseur, C. Demonceaux, and F. Fraundorfer, "A homography formulation to the 3pt plus a common direction relative pose problem," in *Computer vision–ACCV*. Springer, 2014.
[30] D. A. Cox, J. Little, and D. O'Shea, *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra, 3/e (Undergraduate Texts in Mathematics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.
[31] Z. Kukelova, M. Bujnak, and T. Pajdla, "Automatic generator of minimal problem solvers," in *Computer Vision - ECCV 2008, 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part III*, ser. Lecture Notes in Computer Science, D. A. Forsyth, P. H. S. Torr, and A. Zisserman, Eds., vol. 5304. Springer, 2008, pp. 302–315.
[32] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.
[33] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.
[34] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," in *International Journal on Computer Vision (IJCV)*, vol. 60, no. 2, 2004, pp. 91–110.
[35] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.
[36] J. L. Schönberger, F. Fraundorfer, and J.-M. Frahm, "Structure-from-motion for mav image sequence analysis with photogrammetric applications," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL-3, pp. 305–312, 2014. [Online]. Available: http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-3/305/2014/
[37] D. F. M. Rothermel, K. Wenzel and N. Haala, "Sure: Photogrammetric surface reconstruction from imagery," in *Proceedings LC3D Workshop*, 2014.

Olivier Saurer received his MSc in computer science from ETH Zurich in 2009. He is currently a graduate student at ETH Zurich in the Computer Vision and Geometry Group. His research interests include sparse and dense reconstruction methods for omnidirectional and monoclular vision and its application on challenging datasets.

Pascal Vasseur received the M.S. degree in System Control from Université de Technologie de Compiègne (France) in 1995 and his Ph.D. in Automatic Control from Université de Picardie Jules Verne (France) in 1998. He was associate professor at the Université de Picardie Jules Verne in Amiens between 1999 and 2010. He is now a full professor at the Université de Rouen and is member of LITIS laboratory. His research interests are computer vision and its applications to mobile and aerial robots.

Rémi Boutteau received his engineering diploma from the Ecole des Mines de Douai and his MSc degree in computer science and engineering from the University of Science and Technology of Lille (USTL) in 2006. In 2010, he received his PhD degree from the University of Rouen for studies related to computer vision, panoramic vision obtained by catadioptric sensors, and 3D reconstruction algorithms dedicated to omnidirectional vision. After his PhD, he has joined the ESIGELEC engineering school as a lecturer in embedded systems, and the "Instrumentation, Computer Sciences and Systems" research team in the IRSEEM Laboratory. His research interests include computer vision, structure from motion, visual odometry and omnidirectional vision dedicated to autonomous vehicles.

Cédric Demonceaux received the M.S. degree in Mathematics in 2001 and the PhD degree in Image Processing from the Université de Picardie Jules Verne (UPJV), France, in 2004. In 2005, he became associate professor at MIS-UPJV. From 2010 to 2014, he has been an CNRS-Higher Education chair at Le2I UMR CNRS, Université de Bourgogne. Since 2014, he is full Professor at the University of Burgundy. His research interests are in image processing, computer vision and robotics.

Marc Pollefeys is a full professor in the Dept. of Computer Science of ETH Zurich since 2007. Before that he was on the faculty at the University of North Carolina at Chapel Hill. He obtained his PhD from the KU Leuven in Belgium in 1999. His main area of research is computer vision, but he is also active in robotics, machine learning and computer graphics. Dr. Pollefeys has received several prizes for his research, including a Marr prize, an NSF CAREER award, a Packard Fellowship and a European Research Council Grant. He is the author or co-author of more than 250 peer-reviewed publications. He was the general chair of ECCV 2014 in Zurich and program chair of CVPR 2009. He is a fellow of the IEEE.

Friedrich Fraundorfer is Assistant Professor at Graz University of Technology, Austria. He received the Ph.D. degree in computer science from TU Graz in 2006. He had post-doc stays at the University of Kentucky, at the University of North Carolina at Chapel Hill, at ETH Zürich and acted as Deputy Director of the Chair of Remote Sensing Technology at the Technische Universität München from 2012 to 2014. His main research areas are 3D computer vision and robot vision.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2545663, IEEE Transactions on Pattern Analysis and Machine Intelligence

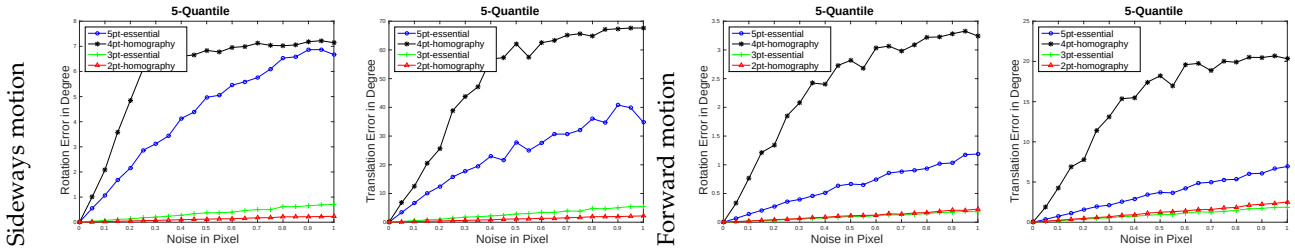

Fig. 3. Evaluation of the 2 point algorithm under sideways and forward motion with varying image noise.

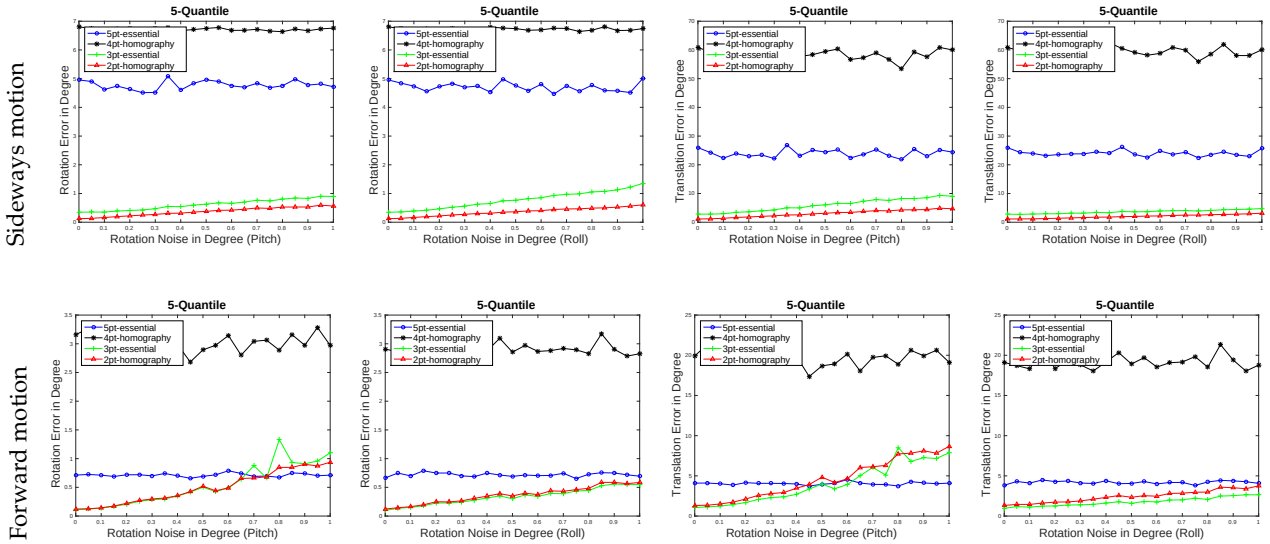

Fig. 4. Evaluation of the 2pt algorithm under different IMU noise and constant image noise of 0.5 pixel standard deviation. First row, sideways motion, second row forward motion.

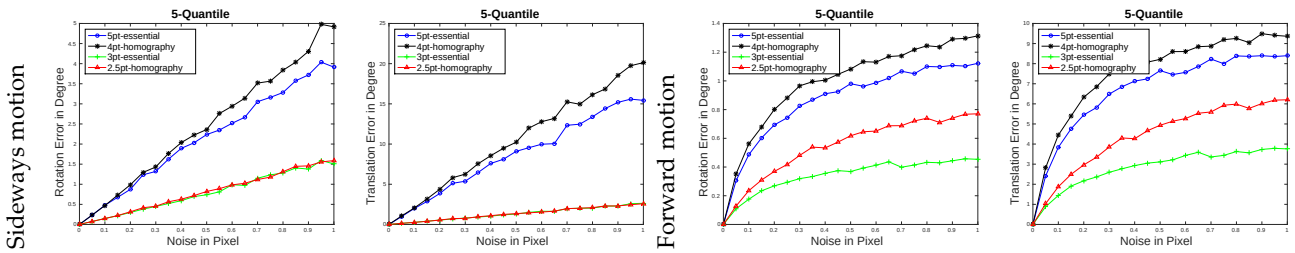

Fig. 5. Evaluation of the 2.5pt algorithm under sideways and forward motion with varying image noise.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2545663, IEEE Transactions on Pattern Analysis and Machine Intelligence

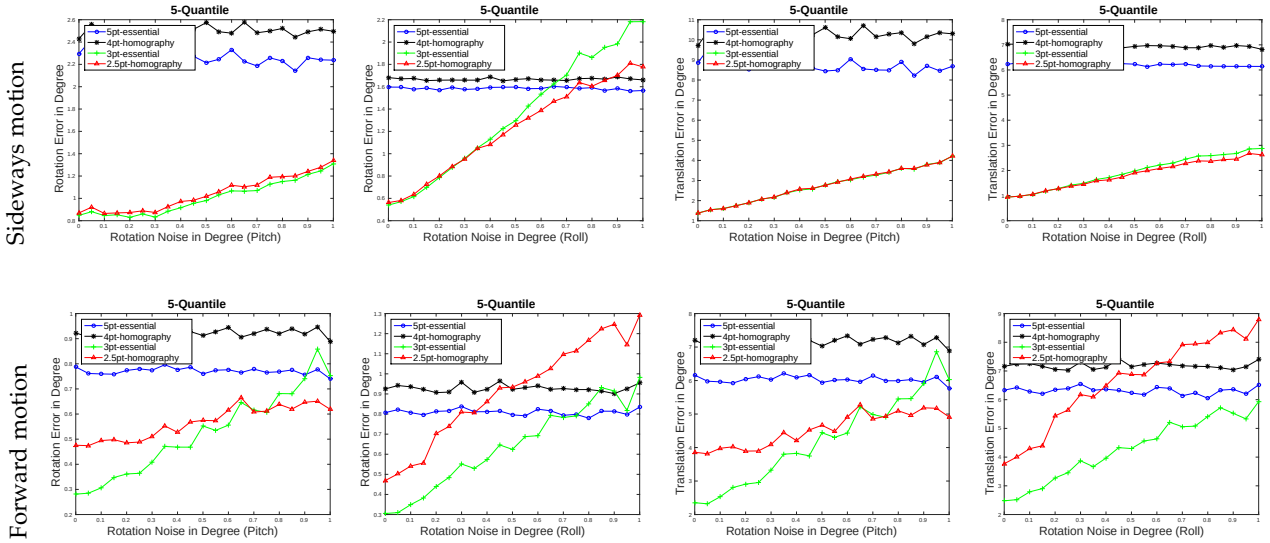

Fig. 6. Evaluation of the 2.5pt algorithm under IMU noise and constant image noise of 0.5 pixel standard deviation. First row sideways motion, second row forward motion.

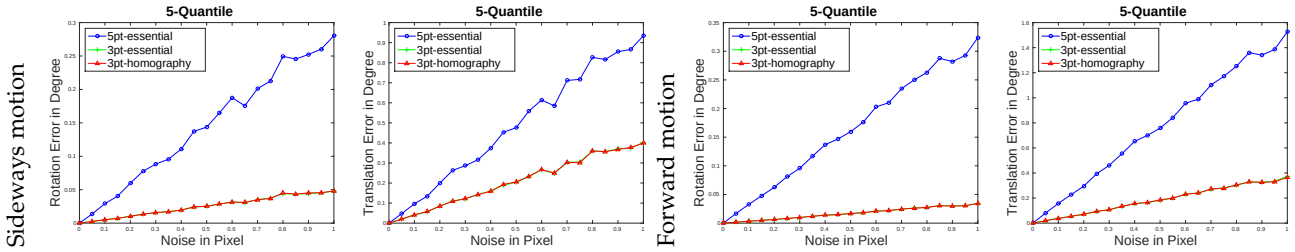

Fig. 7. Evaluation of the 3pt-homography algorithm under sideways and forward motion with varying image noise.

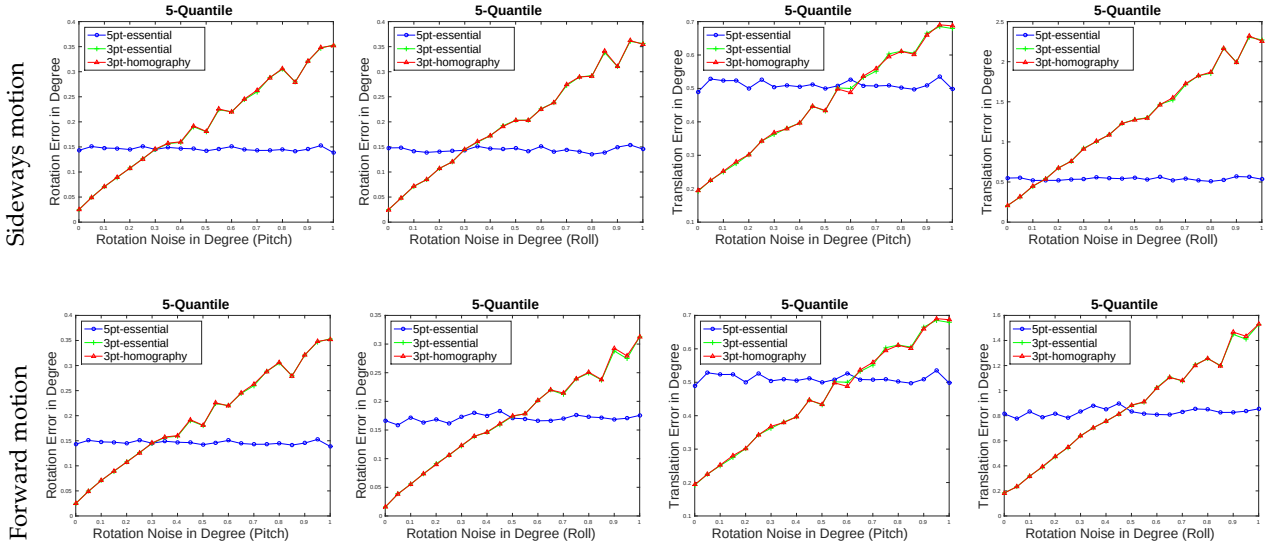

Fig. 8. Evaluation of the 3pt-homography algorithm under different IMU noise and constant image noise of 0.5 pixel standard deviation. First row: sideways motion of the camera with varying pitch angle (left) and varying roll angle (right). Second row: forward motion of the camera with varying pitch angle (left) and varying roll angle (right).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2545663, IEEE Transactions on Pattern Analysis and Machine Intelligence

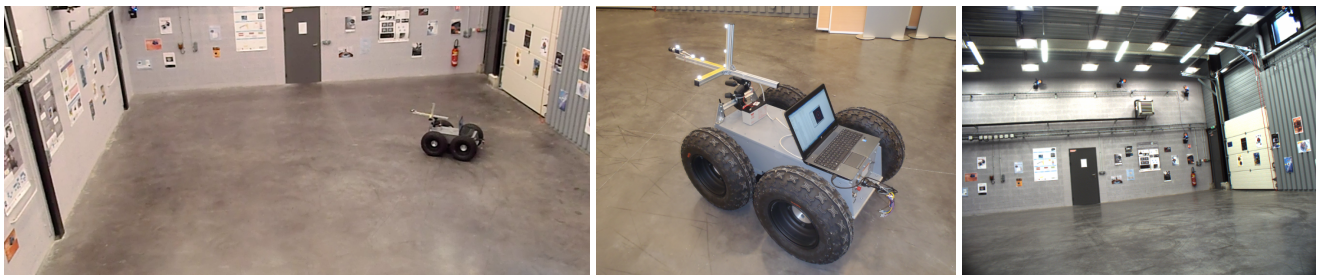

Fig. 9. Left, Vicon arena used to record the ground-truth dataset. Center, teleoperated Segway mobile robot capturing data. Right, sample image captured by the robot.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2545663, IEEE Transactions on Pattern Analysis and Machine Intelligence

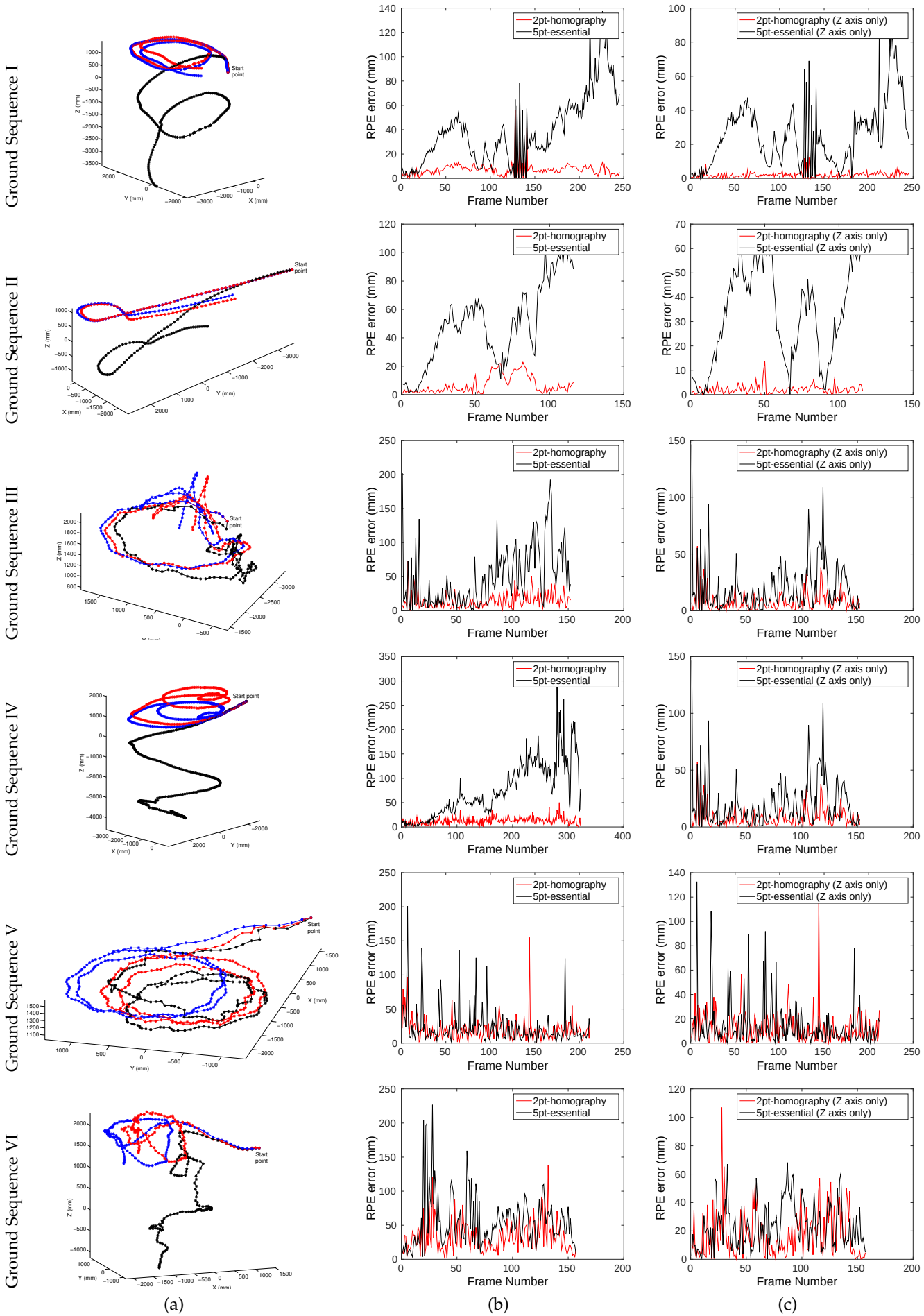

Fig. 10. Evaluation of the 2pt ground plane algorithm. (a) visual odometry estimated using the 2pt (red) and the 5pt (black) algorithm. The Vicon ground-truth is given in blue. (b) the Relative Pose Error (RPE) in *mm* for each individual frame. (c) shows the RPE error for the vertical axis (*z*-axis).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2545663, IEEE Transactions on Pattern Analysis and Machine Intelligence

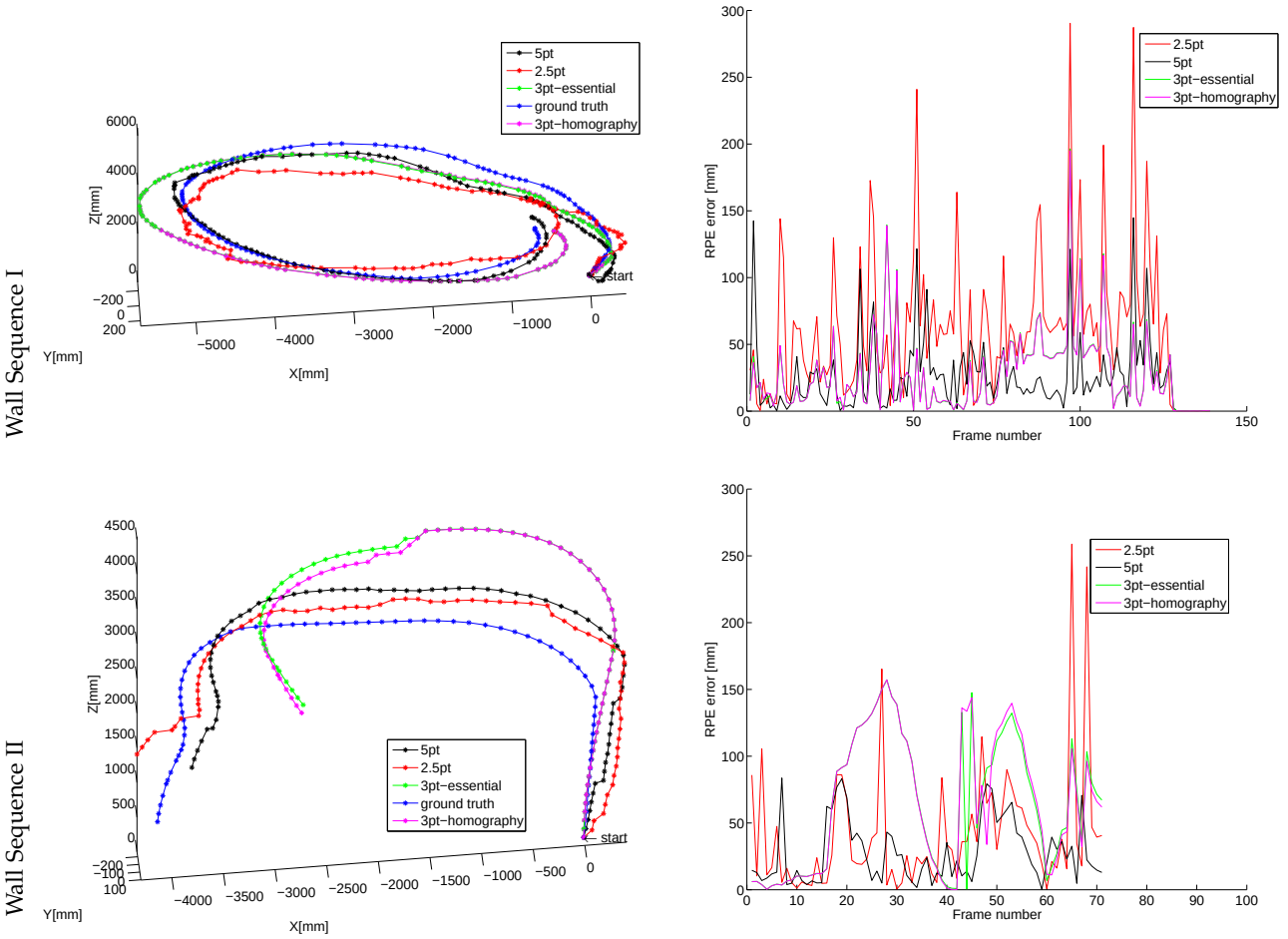

Fig. 11. Evaluation of the 2.5pt vertical wall algorithm: Trajectories estimated with 2.5pt (red), with 3pt-homography (magenta), with 3pt-essential matrix (green) and 5pt (black curve) algorithms compared with the Vicon ground-truth (blue curve).

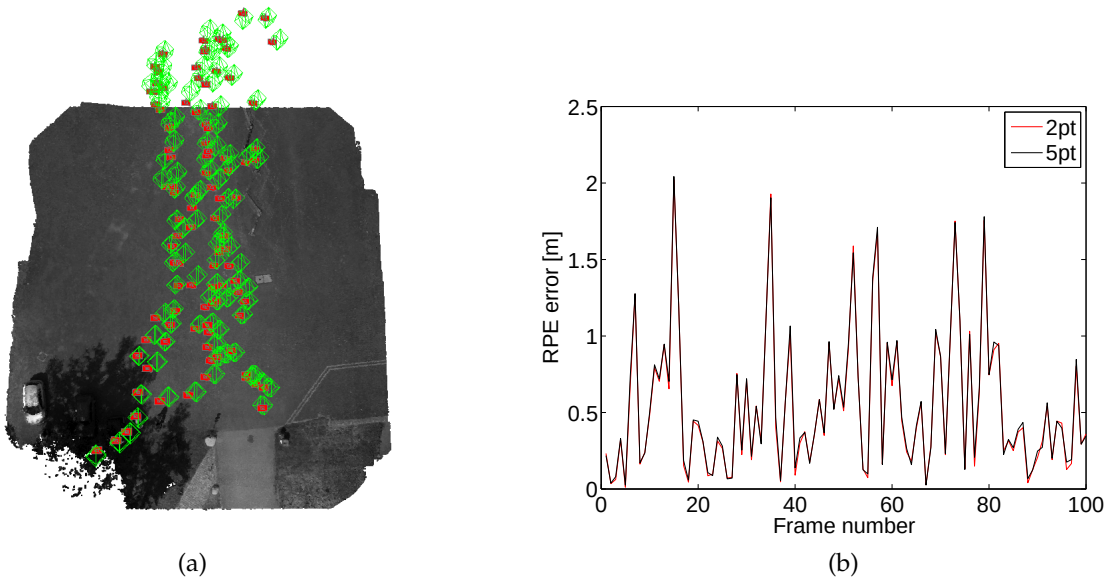

Fig. 12. Results of an incremental SFM pipeline using the 2pt algorithm. (a) Resulting 3D point cloud, camera positions (red) and GPS positions (green). (b) RPE error plot when using 5pt or 2pt within the SFM pipeline. The initial solution of the 5pt and 2pt are similar enough for bundle adjustment to converge to almost the same final solution.