Homography-based Egomotion Estimation Using Gravity and SIFT Features

Yaqing Ding[1], Daniel Barath[2,3], and Zuzana Kukelova[3]

[1] Nanjing University of Science and Technology, China
dingyaqing@njust.edu.cn
[2] Machine Perception Research Laboratory, SZTAKI in Budapest
barath.daniel@sztaki.mta.hu
[3] VRG, Faculty of Electrical Engineering, Czech Technical University in Prague
kukelova@cmp.felk.cvut.cz

Abstract. Camera systems used, e.g., in cars, UAVs, smartphones, and tablets, are typically equipped with IMUs (inertial measurement units) that can measure the gravity vector. Using the information from an IMU, the y-axes of cameras can be aligned with the gravity, reducing their relative orientation to a single DOF (degree of freedom). In this paper, we use the gravity information to derive extremely efficient minimal solvers for homography-based egomotion estimation from orientation- and scale-covariant features. We use the fact that orientation- and scale-covariant features, such as SIFT or ORB, provide additional constraints on the homography. Based on the prior knowledge about the target plane (horizontal/vertical/general plane, w.r.t. the gravity direction) and using the SIFT/ORB constraints, we derive new minimal solvers that require fewer correspondences than traditional approaches and, thus, speed up the robust estimation procedure significantly. The proposed solvers are compared with the state-of-the-art point-based solvers on both synthetic data and real images, showing comparable accuracy and significant improvement in terms of speed. The implementation of our solvers is available at https://github.com/yaqding/relativepose-sift-gravity.

1 Introduction

Estimating the relative camera motion from two views is a fundamental problem in computer vision [1], which usually is approached by applying a minimal solver combined with a robust estimator, e.g., RANSAC [2]. Using a minimal set of point correspondences is important since the processing time of robust estimation depends exponentially on the sample size. The well-known five-point solver [3], which uses only the point coordinate information, is a minimal solution to the relative pose estimation problem with calibrated cameras. In order to reduce the number of necessary points, we usually need to exploit additional prior knowledge about the underlying camera motion or scene geometry. Such a prior is, for example, the assumption that the camera moves on a plane – e.g., it is mounted to a vehicle – and, therefore, only a single rotation and two translation parameters have to be estimated [4, 5]. Recently, largely motivated by the

availability of camera-IMU systems, smart phones and tablets, which have accelerometers to measure the gravity direction, point-plus-direction solvers have shown a number of benefits [6–14]. Using this measurement, the y-axes of the cameras can be aligned with the gravity direction, reducing the relative orientation of two cameras from three to a single degree of freedom (DOF). This allows using only three point correspondences to obtain the relative camera motion (the DOF of the estimated essential matrix reduces from five to three) [7].

Scenes containing large planar surfaces, e.g., floor, walls, doors, street or other general structures, are very common in man-made environments. Thus, homography-based methods [1] also play an important role in relative pose estimation. For a known gravity direction, Saurer et al. [6] show that planes, in such environments, can be divided into three categories: horizontal, vertical and general planes. Using the orientation prior, the number of correspondences required for the estimation is reduced: to two point correspondences for the ground and a vertical plane with known normal, and 2.5 point correspondences for vertical planes with unknown normals. For a general plane, the homography-based relative pose estimation is equivalent to the epipolar geometry (the essential matrix estimation), and therefore, it requires three point correspondences (note that three points always lie on a general plane). In [15], the authors propose a homography-based relative pose estimation approach assuming a known vertical direction with points on the horizontal line.

Affine correspondences encode higher-order information about the underlying scene geometry. Thus, fewer features are needed for model estimation compared to point-based methods. Barath et al. [16, 17] show that two affine correspondences are enough for relative pose estimation when the focal length is fixed and unknown. Recently, it has been shown that the relative pose can be estimated from one affine correspondence with known gravity direction or under the planar motion assumption [18, 19]. However, a major drawback of using such features in practice is that obtaining them, e.g., via Affine-SIFT [20] or Harris-Affine [21], is time-consuming. This severely restricts the applicability of these techniques, especially for real-time applications. Nevertheless, parts of the affine features can be obtained from widely-used feature detectors. For example, SIFT [22] and SURF [23] provide orientation- and scale-covariant features, which allows homography estimation from two correspondences [24]. ORB [25] provides oriented features, and has been successfully used for fundamental matrix estimation [26].

In this paper, we investigate the case where the camera is equipped with an IMU to align the y-axes with the gravity direction. After aligning the camera coordinate system, we may have some information about the normal of the target plane, e.g., the ground plane usually becomes parallel to the XZ plane of the aligned system. Since the scale and rotation of the feature is known at no cost when using most of the widely-used feature detectors, e.g., SIFT or SURF, we propose minimal solvers for homography estimation in man-made environments based on orientation- and scale-covariant features. Our work builds on top of the work of Saurer et al. [6] and Barath et. al. [24]. Our new solvers exploit the orientation and scale constraints on Euclidean homographies introduced in [24]

and apply them for homography-based egomotion estimation with a known vertical direction. In contrast to the point correspondences used in the solvers of Saurer et al. [6], our solvers require fewer correspondences. Thus, the new solvers speed up the robust estimation procedure significantly. We support this claim by extensive experiments, where we show that the new solvers have comparable accuracy and notable improvement in speed over point-based solvers [6, 3].

The main **contributions** of this paper are: (i) Since the affine transformations will change after the cameras are aligned with the gravity direction, the scale and orientation from features detected in the original image cannot be directly used for the aligned cameras. We thus investigate the relationship between the original and aligned views so that the scale and orientation from the original views can give constraints on the aligned views. (ii) When the points lie on the ground plane, we show that the relative pose problem can be solved from a single orientation- and scale-covariant feature. (iii) In addition, we prove that the rotation estimation is independent of the feature scale for this case. (iv) If the points are on a vertical plane, we show that a single orientation-covariant feature and one point correspondence are sufficient to estimate the camera motion. (v) In the case that the normal of the plane is completely unknown, we derive a minimal solution using only two orientation-covariant features for general homography estimation. For more details on this solver we refer the reader to the supplementary material.

2 Background

Suppose that we are given a point on a 3D plane and its two projections, $\mathtt{m}_1 = [u_1, v_1, 1]^\top$ and $\mathtt{m}_2 = [u_2, v_2, 1]^\top$, with respect to two camera frames. These two image points are related by a Euclidean homography matrix $\mathtt{H} \in \mathbb{R}^{3\times3}$ as

$$\gamma \mathtt{K}_2^{-1}\mathtt{m}_2 = \mathtt{H}\mathtt{K}_1^{-1}\mathtt{m}_1 \ , \tag{1}$$

where $\gamma \in \mathbb{R}$ is a scaling factor, and $\mathtt{K}_1, \mathtt{K}_2 \in \mathbb{R}^{3\times3}$ are the camera intrinsic matrices. The homography matrix $\mathtt{H}$ relates the normalized image points $\mathtt{K}_1^{-1}\mathtt{m}_1$ and $\mathtt{K}_2^{-1}\mathtt{m}_2$. $\mathtt{H}$ is related to the rotation $\mathtt{R}$, translation $\mathtt{T}$, and distance d from the camera frame to the target plane, and the normal $\mathtt{N}$ of the plane via

$$\mathtt{H} = (\mathtt{R} + \mathtt{T}\mathtt{N}^\top) \ , \tag{2}$$

where we can absorb the distance d into the translation $\mathtt{T}$. Here, we assume that the gravity direction is known. The gravity direction can be calculated, e.g., from the IMU data or vanishing points [1]. With this assumption, and without loss of generality, we can rotate the points so that the y-axes of the cameras are aligned with the gravity direction (Fig. 1(a)). Let $\mathtt{R}_1, \mathtt{R}_2$ be the rotation matrices that were used for the alignment of the first and the second camera. Applying the rotations to the normalized image points, Eq. (1) becomes

$$\gamma \mathtt{R}_2\mathtt{K}_2^{-1}\mathtt{m}_2 = \mathtt{H}_y\mathtt{R}_1\mathtt{K}_1^{-1}\mathtt{m}_1 \ , \tag{3}$$

with

$$\mathtt{H}_y = (\mathtt{R}_y + \mathtt{t}\mathtt{n}^\top) \ . \tag{4}$$

Here, $\mathtt{t}$ is the translation (the distance is absorbed into $\mathtt{t}$) after the alignment and $\mathtt{R}_y$ is an unknown rotation around the y-axis of the form

$$\mathtt{R}_y = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \ , \tag{5}$$

where θ is the unknown yaw angle. The relationship between $\mathtt{H}$ and $\mathtt{H}_y$ is given by the following formula:

$$\mathtt{H} = \mathtt{R}_2^\top \mathtt{H}_y \mathtt{R}_1 \ . \tag{6}$$

Hence, the full relative rotation and translation between the two views are

$$\mathtt{R} = \mathtt{R}_2^\top \mathtt{R}_y \mathtt{R}_1 \ , \quad \mathtt{T} = \mathtt{R}_2^\top \mathtt{t} \ . \tag{7}$$

In this case, $\mathtt{H}_y$ can be used to get the 5-DOF relative pose between the views.

2.1 DOF analysis

It is well-known that the general Euclidean homography matrix has 8 degrees of freedom originating from the three parameters of rotation $\mathtt{R}$, two unknowns describing the orientation of the unit plane normal $\mathtt{n}$, the distance d of the plane from the camera, and two DOF from the direction of $\mathtt{t}$.

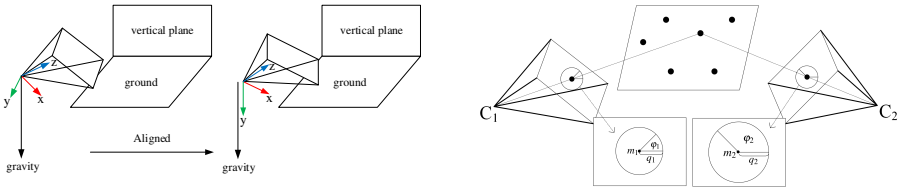
In our case, when the coordinate system is aligned, the DOF of $\mathtt{R}$ is reduced to one, i.e., the angle of rotation around the vertical axis. In general configurations, the relative translation $\mathtt{t}$ between the two cameras has only two degrees of freedom since it can only be recovered up to scale, due to the nature of perspective projection [1]. However, since the distance d of the observed plane is absorbed into $\mathtt{t}$ to achieve the special form of $\mathtt{H}_y$, $\mathtt{t}$ cannot be arbitrarily scaled and, thus, it has three degrees of freedom. As a consequence, we will be estimating all three parameters of $\mathtt{t}$ and, also, the unknown rotation angle.

2.2 Orientation- and scale-covariant feature constraints

In this paper, we use orientation- and scale-covariant features to estimate the unknown homography. Most of the widely-used feature detectors, e.g., SIFT and SURF, not only provide point correspondences but also additional information about each feature's scale and rotation. This means that an orientation- and scale-covariant feature correspondence can be considered as a correspondence of triplets $(\mathtt{m}_1, \varphi_1, q_1) \leftrightarrow (\mathtt{m}_2, \varphi_2, q_2)$, where $\mathtt{m}_1 \leftrightarrow \mathtt{m}_2$ is a point correspondence, and φ_i and q_i, $i = 1, 2$ are, respectively, the orientation and the scale of the feature.

For the point correspondence part, Eq. (1) holds and can be rewritten as

$$\gamma \mathtt{m}_2 = \mathtt{G}\mathtt{m}_1 \ , \tag{8}$$



(a) (b)

Fig. 1: (a) Illustration of the coordinate systems used in this paper. We can align the y-axis of the camera with the gravity based on the IMU readings. (b) Visualization of orientation- and scale-covariant features.

where $\mathtt{G}$ is a homography matrix which relates uncalibrated image points $\mathtt{m}_1$ and $\mathtt{m}_2$, i.e., $\mathtt{G} = \mathtt{K}_2\mathtt{H}\mathtt{K}_1^{-1}$. This means that each point correspondence yields the following two linear constraints on $\mathtt{G}$:

$$\begin{bmatrix} 0 & 0 & 0 & -u_1 & -v_1 & -1 & v_2u_1 & v_2v_1 & v_2 \\ u_1 & v_1 & 1 & 0 & 0 & 0 & -u_2u_1 & -u_2v_1 & -u_2 \end{bmatrix} \mathtt{g} = \mathtt{0} \ , \tag{9}$$

$$\mathtt{g} = [g_1 \ g_2 \ g_3 \ g_4 \ g_5 \ g_6 \ g_7 \ g_8 \ g_9]^\top \ , \tag{10}$$

where $g_1, g_2, ..., g_9$ are the elements of the homography $\mathtt{G}$ in a row-major order.

In [24], two constraints that relate the homography matrix to the scales and rotations of the orientation- and scale-covariant features, e.g., SIFTs, in the first and second image are derived. These constraints have the form

$$\begin{aligned} g_8u_2s_1s_2 + g_7u_2s_2c_1 - g_8v_2s_1c_2 - g_7v_2c_1c_2 + \\ -g_2s_1s_2 - g_1s_2c_1 + g_5s_1c_2 + g_4c_1c_2 = 0 \ , \end{aligned} \tag{11}$$

$$\begin{aligned} g_7^2u_1^2q_2 + 2g_7g_8u_1v_1q_2 + g_8^2v_1^2q_2 + g_5g_7u_2q_1 + \\ -g_4g_8u_2q_1 - g_2g_7v_2q_1 + g_1g_8v_2q_1 + 2g_7g_9u_1q_2 + \\ 2g_8g_9v_1q_2 + g_2g_4q_1 - g_1g_5q_1 + g_9^2q_2 = 0 \ , \end{aligned} \tag{12}$$

where $c_i = \cos(\varphi_i)$, $s_i = \sin(\varphi_i)$. Note that the first constraint (11) is not dependent on scales. It can thus also be used for features that are orientation- but not scale-covariant, e.g., ORB features [25].

This means that each SIFT correspondence gives four constraints on the homography matrix $\mathtt{G}$ (three linear ones from (9) and (11) and one quadratic one from (12)) and each ORB correspondence gives just three linear constraints ((9) and (11)). Our objective is to estimate the camera motion by estimating a homography from a combination of SIFT/ORB and point correspondences. Note, that constraints (11) and (12) were derived for a general homography. In our case, the y-axes of the cameras are aligned with the gravity direction. This changes not only the point coordinates, but also affects the orientation and scale of the features. Thus, the constraints (11) and (12) cannot be directly applied when estimating the Euclidean homography matrix $\mathtt{H}_y$ via Eq. (3). Next, we describe how to use the orientation and scale constraints to estimate the homography $\mathtt{H}_y$ under different assumptions about the camera motion and observed plane.

3 Pose Estimation

Based on (1), (6) and (8), the relationship between the Euclidean homography matrix $\mathtt{H}_y$ and the standard homography matrix $\mathtt{G}$ can be formulated as

$$\mathtt{H}_y \sim \widehat{\mathtt{H}}_y = \mathtt{R}_2\mathtt{K}_2^{-1}\mathtt{G}\mathtt{K}_1\mathtt{R}_1^\top \ , \tag{13}$$

where $\sim$ indicates equality up to a scaling factor, i.e., $\widehat{\mathtt{H}}_y = \lambda\mathtt{H}_y$, for some scale $\lambda \neq 0$. In order to find the constraints on the Euclidean homography matrix $\mathtt{H}_y$, we first consider the matrix $\mathtt{B}$ defined as

$$\mathtt{B} = \widehat{\mathtt{H}}_y - \lambda\mathtt{R}_y = \lambda\mathtt{t}\mathtt{n}^\top \ . \tag{14}$$

$\mathtt{B}$ is the difference between the homography $\widehat{\mathtt{H}}_y$ and the corresponding rotation $\mathtt{R}_y$ (we have to add a scalar λ because (13) holds up to scale). Matrix $\lambda\mathtt{R}_y$ can be written as

$$\lambda\mathtt{R}_y = \begin{bmatrix} \alpha & 0 & \beta \\ 0 & \lambda & 0 \\ \text{-}\beta & 0 & \alpha \end{bmatrix} \ , \tag{15}$$

where $\alpha = \lambda\cos\theta$, $\beta = \lambda\sin\theta$, and θ is the yaw angle associated with $\mathtt{R}_y$. Matrix $\mathtt{B}$ in (14) can then be rewritten as

$$\mathtt{B} = \begin{bmatrix} h_1 - \alpha & h_2 & h_3 - \beta \\ h_4 & h_5 - \lambda & h_6 \\ h_7 + \beta & h_8 & h_9 - \alpha \end{bmatrix} \ , \tag{16}$$

where h_i are the elements of the matrix $\widehat{\mathtt{H}}_y$.

Next, we consider two different cases based on prior knowledge of the type of target plane, i.e., situations where our points lie on a horizontal or a vertical plane. The third case, i.e., a general plane, is discussed in the supplementary material. The prior knowledge of the type of target plane gives additional constraints on the form of the matrix $\mathtt{B}$ that simplify the computation of $\mathtt{H}_y$.

3.1 Points on a Horizontal Plane

An important case arises when the points lie on a horizontal plane. This is practical and relevant when a camera is mounted on a vehicle/robot or a UAV mounted with a bird-view camera, since the ground plane is virtually always available. In this case, $n = [0\ 1\ 0]^\top$, and we have a 4-DOF problem with respect to $[\theta, t_x, t_y, t_z]$. Hence, this problem can be solved from a single SIFT correspondence which yields four independent constraints. From Eq. (14), it follows that the matrix $\mathtt{B}$ has the form

$$\mathtt{B} = \lambda \begin{bmatrix} 0 & t_x & 0 \\ 0 & t_y & 0 \\ 0 & t_z & 0 \end{bmatrix} \ . \tag{17}$$

Using (16) and (17), we obtain six constraints on the elements of the homography matrix $\widehat{\mathtt{H}}_y$, i.e.,

$$h_4 = 0,\ h_6 = 0,\ h_1 - \alpha = 0,\ h_3 - \beta = 0,\ h_9 - \alpha = 0,\ h_7 + \beta = 0\ . \tag{18}$$

After eliminating the parameters $\{\alpha, \beta\}$ based on the last four equations in (18), we obtain four constraints

$$h_4 = 0, \quad h_6 = 0, \quad h_1 - h_9 = 0, \quad h_3 + h_7 = 0 \tag{19}$$

which generally hold for this type of homographies. Given a single SIFT correspondence, we have three linear constraints (from (9) and (11)) on the homography matrix $\mathtt{G}$. These constraints can be written in a matrix form

$$\mathtt{M}\mathtt{g} = \mathtt{0}\ , \tag{20}$$

where $\mathtt{M}$ is a 3×9 coefficient matrix and the vector $\mathtt{g}$ contains the elements of $\mathtt{G}$ (cf. (10)). Vector $\mathtt{g}$ can be written as a linear combination of the six basis vectors from the 6-dimensional null space of the matrix $\mathtt{M}$ as

$$\mathtt{g} = x_1\mathtt{g}_a + x_2\mathtt{g}_b + x_3\mathtt{g}_c + x_4\mathtt{g}_d + x_5\mathtt{g}_e + x_6\mathtt{g}_f\ , \tag{21}$$

where $x_1, \ldots, x_6$ are new unknowns. Note that since $\mathtt{G}$ is given only up to scale, we can fix one of the unknowns, e.g., $x_6 = 1$. Substituting (21) into (13) yields $\widehat{\mathtt{H}}_y = \mathtt{R}_2\mathtt{K}_2^{-1}\mathtt{G}\mathtt{K}_1\mathtt{R}_1^\top$. The Euclidean homography matrix $\widehat{\mathtt{H}}_y$, for calibrated cameras, can be parameterized using five unknowns $\{x_1, x_2, x_3, x_4, x_5\}$. Since there are four linear constraints (19) on the elements of $\widehat{\mathtt{H}}_y$, we can use these ones to express four from the five unknowns, e.g., $x_1, \ldots, x_4$ as a linear functions of x_5. This leads to a parameterization of the homography matrix $\mathtt{G}$ (21), as well as $\widehat{\mathtt{H}}_y$, using just one unknown, i.e., x_5. This parameterization is finally substituted into the SIFT constraint (12), leading to one quadratic equation in x_5. After solving this equation, $\widehat{\mathtt{H}}_y$ and, subsequently, $\mathtt{R}_y$ and $\mathtt{t}$ can be recovered. Although we can directly decompose $\mathtt{G}$ into $\mathtt{R}$ and $\mathtt{T}$ using the SVD-based method, there will be two possible rotations for a standard homography matrix $\mathtt{G}$. By contrast, each $\widehat{\mathtt{H}}_y$ corresponds to a unique rotation, where the redundant solution is eliminated.

3.2 Points on a Vertical Plane

As a complement to the case where points are on a horizontal plane, we address the case where they lie on a vertical plane with unknown normal. This is also practical, since walls and building facades in man-made environments are usually parallel to the gravity direction. In this case, $\mathtt{n} = [n_x\ 0\ n_z]^\top$ and we have a 5-DOF problem w.r.t. $\{\theta, t_x, t_y, t_z, n_x, n_z\}$ (the unit vector $\mathtt{n}$ has two parameters and one DOF). We need at least one SIFT or ORB and one point correspondences to solve this problem compared to the 2.5 point correspondences of the point-based solver [6]. Note that one SIFT and one point correspondence provide 6 independent constraints which result in an over-constrained system. We choose to only

use the orientation constraint (11) from the feature (SIFT/ORB) correspondence since, in practice, the scale is usually more noise-sensitive. Moreover, the orientation constraint (11) is linear compared to the quadratic constraint (12). The new solver only requires oriented features and thus also works with ORB [25] features. Under the vertical plane constraint, matrix $\mathtt{B}$ in (14) becomes

$$\mathtt{B} = \lambda \begin{bmatrix} t_x n_x & 0 & t_x n_z \\ t_y n_x & 0 & t_y n_z \\ t_z n_x & 0 & t_z n_z \end{bmatrix} \ . \tag{22}$$

There are 6 constraints on the matrix $\mathtt{B}$ as follows:

$$\begin{aligned} \mathtt{B}(1,2) = 0,\ \mathtt{B}(2,2) = 0,\ \mathtt{B}(3,2) = 0 \ , \\ \mathtt{B}(1,1)\mathtt{B}(2,3) = \mathtt{B}(1,3)\mathtt{B}(2,1) \ , \\ \mathtt{B}(1,1)\mathtt{B}(3,3) = \mathtt{B}(1,3)\mathtt{B}(3,1) \ , \\ \mathtt{B}(2,1)\mathtt{B}(3,3) = \mathtt{B}(2,3)\mathtt{B}(3,1) \ . \end{aligned} \tag{23}$$

Substituting (16) into (23), we have the following 6 equations

$$\begin{aligned} h_2 = 0,\ h_8 = 0,\ h_5 - \lambda = 0,\ h_4(h_9 - \alpha) - h_6(h_7 + \beta) = 0 \ , \\ h_6(h_1 - \alpha) - h_4(h_3 - \beta) = 0,\ (h_1 - \alpha)(h_9 - \alpha) - (h_3 - \beta)(h_7 + \beta) = 0 \ . \end{aligned} \tag{24}$$

Similar to the case where points are on a horizontal plane, we find that two of the equations in (24) (forth and fifth) are linear in α and β. Thus, we can use them to express α and β using h_i. Substituting the formulation into the last equation of (24), we obtain the following constraint without parameters $\{\lambda, \alpha, \beta\}$:

$$(h_1 h_6 - h_3 h_4)^2 + (h_4 h_9 - h_6 h_7)^2 - h_5^2(h_4^2 + h_6^2) = 0 \ . \tag{25}$$

Together with $h_2 = 0,\ h_8 = 0$, we obtain three constraints for the vertical plane-induced Euclidean homography matrix H_y. Given one SIFT/ORB and one point correspondences, we have 3+2 linear constraints ((9) and (11)) on the homography matrix $\mathtt{G}$. These constraints can be written in a matrix form

$$\mathtt{M}\mathtt{g} = \mathtt{0} \ , \tag{26}$$

where $\mathtt{M}$ is a 5×9 coefficient matrix and the vector $\mathtt{g}$ contains the elements of $\mathtt{G}$. The vector $\mathtt{g}$ can be written as a linear combination of the four basis vectors from the 4-dimensional null space of the matrix $\mathtt{M}$ as

$$\mathtt{g} = x_1 \mathtt{g}_a + x_2 \mathtt{g}_b + x_3 \mathtt{g}_c + x_4 \mathtt{g}_d \ , \tag{27}$$

where x_1, x_2, x_3, x_4 are new unknowns. Again, we fix one of the unknowns, e.g., $x_4 = 1$. Substituting (27) into (13) yields $\widehat{\mathtt{H}}_y = \mathtt{R}_2 \mathtt{K}_2^{-1} \mathtt{G} \mathtt{K}_1 \mathtt{R}_1^\top$. The Euclidean homography matrix $\widehat{\mathtt{H}}_y$, for calibrated cameras, can be parameterized using three unknowns $\{x_1, x_2, x_3\}$. We can use $h_2 = 0,\ h_8 = 0$ to express two from these three unknowns, e.g., x_1, x_2 as a linear functions of x_3. This leads to a parameterization of the homography matrix $\mathtt{G}$, as well as $\widehat{\mathtt{H}}_y$, using just one unknown x_3. This parameterization is finally substituted into the Euclidean homography constraint (25), leading to one quartic equation in x_3. The remaining steps are the same as for the case where points lie on the horizontal plane.

Table 1: Theoretical computational complexity of solvers (gray – proposed).

Solver	SVD	QR	Eigen	Operations	Total operations with outliers 0.25	0.50	0.75	0.90
1SIFT Ground	3×9	4×4	-	145	482	964	2321	6338
2PC Ground [6]	4×5	-	-	80	446	1281	5709	$3.6*10^4$
1SIFT+1PC Vert.	5×9	2×2	4×4	297	1655	4755	$2.1*10^4$	$1.3*10^5$
3PC Vertical [6]	5×6	-	4×4	214	1799	7381	$6.2*10^4$	$9.8*10^5$
3PC Essential [7]	3×6	6×6	4×4	334	2807	$1.1*10^4$	$9.7*10^4$	$1.5*10^6$
5PC [3]	5×9	10×10	10×10	2225	$3.8*10^4$	$3.2*10^5$	$\sim 10^7$	$\sim 10^9$

4 Experiments

In this section, we study the performance of our solvers on both synthetic data and real images. We compare our solvers with the most closely related work by Saurer et al. [6] and the three-point essential matrix-based solver [7]. Since in [6], the authors have shown that the five-point essential matrix-based solver [3] outperforms the four-point homography algorithm [1], we only compare with the five-point algorithm. Note that there are several different solvers for the five-point relative pose problem [27–29] and three-point relative problem [7, 9, 30, 11, 12, 14], respectively. However, solving the same problem from the same data using stable solvers, the results should be almost equal. We thus only compare against one solver per problem.

4.1 Computational Complexity

Table 1 shows the theoretical computational complexity of each solver. The number of operations for one RANSAC iteration for each solver is calculated based on the solvers' major computations, including SVD, QR and Eigenvalue decomposition. The total number of operations (last four columns) are given as the number of operations for one iteration multiplied by the number of RANSAC iterations for different percentage of outliers. We can see that our solvers need significant fewer operations than the other techniques.

4.2 Synthetic Evaluation

We chose the following setup that is similar to [6] to generate synthetic data. 100 spatial points were distributed uniformly on two planes, a horizontal plane and a vertical plane. The focal length f_g of the camera was set to 500 pixels and the resolution of the image to 1000×1000 pixels. We focused on two practical motions: forward motion (along the z-axis) and sideways motion (along the x-axis). The Euclidean distance between the two cameras was set to be 10 percent of the average scene distance. In addition, the two cameras were rotated

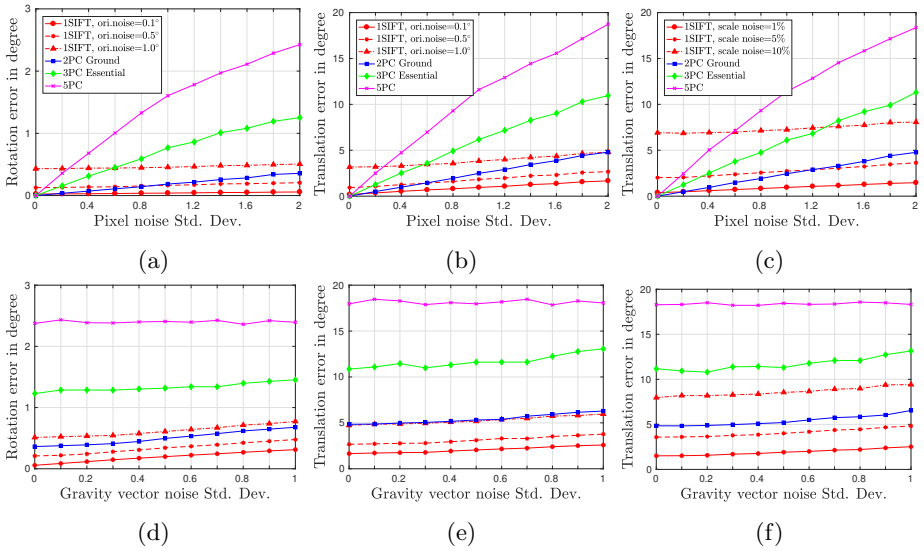


Fig. 2: Comparing the proposed 1SIFT solver with the 2PC Ground solver [6], 3PC Essential solver [7] and the 5PC solver [3]. **Top**: Increasing image noise. (a,b) Rotation and translation errors with additional noise added to SIFT orientations. (c) Translation error with additional noise added to SIFT scales. **Bottom**: Increasing gravity vector noise and fixed image noise (2 px std.). (d,e) Rotation and translation errors with noise added to SIFT orientations. (f) Translation error with noise added to SIFT scales.

around every axis. It was guaranteed that each 3D point was observed by two cameras – however, it is theoretically not required. This is similar to [7, 6]. We generated 10,000 pairs of images with different transformations. To simulate the SIFT orientations and scales, the affine transformations were calculated from the homography using the 4-point algorithm. Then the affine transformations were decomposed into SIFT orientations and scales based on [24]. The rotation error was defined as the angle difference between the estimated rotation and the true rotation: $\arccos\left(\left(\mathrm{tr}(\mathtt{R}_g\mathtt{R}_e^\top)-1\right)/2\right)$, where $\mathtt{R}_g$ and $\mathtt{R}_e$ represent the true and estimated rotations, respectively. The translation error was measured as the angle between the estimated and true translation vectors, since the estimated translation is only defined up to scale. We only show the results for forward motion. Results for sideways motion are shown in the supplementary material.

Fig. 2 reports the rotation and translation errors for points on the ground plane. The top row shows the performance under image noise with different standard deviations. We also add different levels of noise to the SIFT orientations and scales (see the legend). We ran 10,000 trials per data point in the plots and the first quartile of the rotation and translation errors are plotted. This measure is an appropriate performance criterion when the method is used for RANSAC or other robust statistics [6]. The bottom row shows the performance with increased

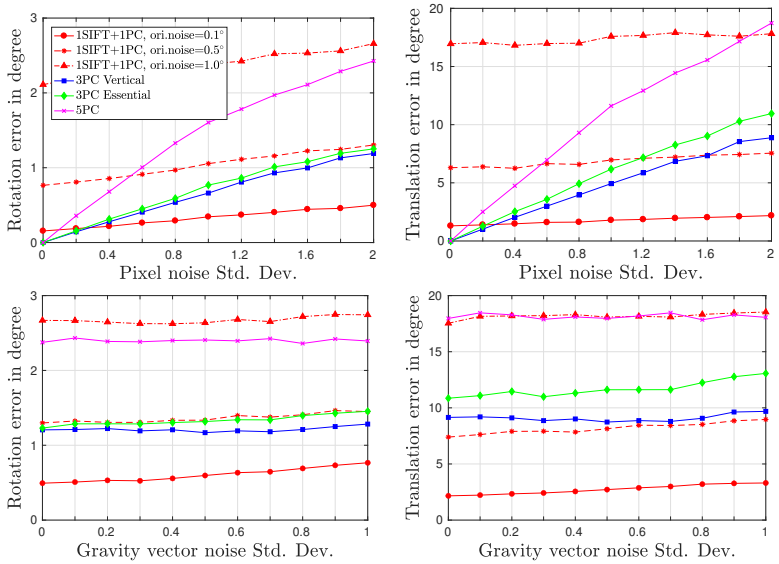
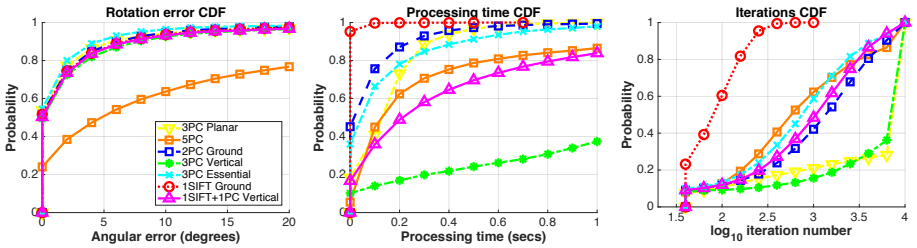


Fig. 3: Comparing the proposed 1SIFT+1PC solver with 3PC Vertical [6], 3PC Essential [7] and 5PC [3]. **Top**: Rotation and translation errors under image noise. **Bottom**: Rotation and translation errors under gravity vector noise and fixed image noise (2 px std.).

gravity vector noise and constant image noise of 2 pixel standard deviation. As we can see, for different levels of noise, our 1SIFT solver performs well and obtains promising results. As shown in previous studies, e.g., [7], smartphones in 2009 such as Nokia N900 and iPhone 4 had a maximum gravity vector error of 1°, which is the maximum tested value in our synthetic experiments. For 1°, the new solvers provide stable results and thus, are useful even for low-cost sensors. Nowadays, accelerometers used in cars and modern smartphones have noise levels around 0.06° (and expensive "good" accelerometers have $< 0.02°$) [7].

Note that, we do not report the rotation error under SIFT scale noise, since there is a special property for our 1SIFT solver: the rotation estimation is independent of the scale. The proof is given in the supplementary material.

Fig. 3 reports the rotation and translation errors for points on a vertical plane. The top row shows the performance under image noise with different standard deviations. Since this solver does not use the scale information from SIFT, we only need to add noise to the SIFT orientations. The bottom row shows the performance with increased gravity vector noise and constant image noise of 2 pixel standard deviation. In this case, our vertical plane-based solver is slightly sensitive to orientation noise. However, with the image noise increased, it is still comparable to other solvers.



(a) (b) (c)

Fig. 4: The cumulative distribution function of the rotation errors; processing time; and $\log_{10}$ iteration number of GC-RANSAC. The values were calculated from a total of 9,064 image pairs (15 scenes from the Malaga dataset). Being more accurate or faster is interpreted as a curve close to the top-left corner.

4.3 Camera mounted to a moving vehicle

In order to test the proposed technique on real-world data, we chose the `Malaga`[4] dataset [31]. This dataset was gathered entirely in urban scenarios with car-mounted sensors, including one high-resolution stereo camera and five laser scanners. We used the sequences of one high-resolution camera and every 10th frame from each sequence. The proposed method was applied to every consecutive image pair. The ground truth paths were composed using the GPS coordinates provided in the dataset. Each consecutive frame-pair was processed independently and we did not run any optimization minimizing the error on the whole path or detecting loop-closures. The estimated relative poses of the consecutive frames were simply concatenated. In total, 9,064 image pairs were used.

Considering that the orientation and scale of local features are often noisier than their point coordinates, we chose to use a locally optimized RANSAC, i.e., Graph-Cut RANSAC[5] [32] (GC-RANSAC), as the robust estimator, where the local optimization is applied to only the point coordinates, similarly as in [24]. In GC-RANSAC (and other RANSAC-like methods), two different solvers are used: (a) one for fitting to a minimal sample and (b) one for fitting to a non-minimal sample when doing model polishing on all inliers or in the local optimization step. For (a), the main objective is to solve the problem using as few data points as possible since the processing time depends exponentially on the number of points required for the model estimation. The proposed and compared solvers were included in this part of the robust estimator. Also, we observed that the considered special planes usually have lower inlier ratio, being localized in the image, compared to general ones. Therefore, instead of verifying the homography in the RANSAC loop, we composed the essential matrix immediately from the recovered pose parameters and did not use the homography itself. For (b), we applied the planar 3PC essential matrix solver [7] to estimate the relative pose

[4] https://www.mrpt.org/MalagaUrbanDataset
[5] https://github.com/danini/graph-cut-ransac

Table 2: The avg. angular error (degrees) and processing time (seconds) of the pose estimation on 13 scenes (columns) of the Malaga dataset using different minimal solvers and GC-RANSAC as robust estimator [32]. The compared methods are the planar solver of [33] (3PC), essential matrix solver of [7] (3PC Ess.), solver of [3] (5PC), ground (2PC Ground) and vertical plane (3PC Vert.) solvers of [6], and the proposed two SIFT-based solvers assuming points on the ground (1SIFT) or vertical plane (1SIFT+1PC). The CDFs are in shown Fig. 4.

	1	2	3	4	5	6	7	8	9	10	11	12	13	avg
	Angular error (°)													
3PC	2.6	3.4	2.9	4.6	3.8	**4.2**	3.9	1.8	4.4	6.3	2.8	2.8	6.2	3.6
3PC Ess.	**2.4**	**2.5**	**1.3**	**1.9**	**3.2**	**4.2**	**3.5**	**1.6**	**3.2**	**5.8**	**1.4**	**2.7**	**5.0**	**2.7**
5PC	5.5	14.8	18.6	18.1	10.7	9.3	13.2	18.0	22.0	13.1	18.2	13.2	24.9	16.5
2PC Ground	2.5	3.0	2.2	2.1	3.8	**4.2**	4.1	2.5	4.3	6.8	2.1	2.8	5.7	3.5
3PC Vert.	2.5	3.6	4.5	3.5	4.1	4.4	4.3	2.9	8.5	5.9	2.5	3.9	6.9	4.4
1SIFT+1PC	2.5	3.6	4.0	3.1	4.1	4.3	4.2	3.1	5.2	7.0	3.2	3.5	6.7	4.2
1SIFT	**2.4**	3.5	2.0	2.5	4.0	4.3	**3.5**	2.1	3.9	6.3	2.2	3.2	6.4	3.5
	Time (seconds)													
3PC	0.20	0.22	0.15	0.19	0.21	0.15	0.18	0.24	0.17	0.20	0.14	0.19	0.24	0.19
3PC Ess.	0.17	0.27	0.06	0.20	0.22	0.09	0.12	0.16	0.08	0.24	0.09	0.10	0.21	0.19
5PC	0.28	0.48	0.26	0.76	0.29	0.27	0.21	0.23	0.10	0.22	0.50	0.18	0.41	0.31
2PC Ground	0.06	0.14	0.03	0.11	0.17	0.03	0.09	0.07	0.08	0.21	0.05	0.09	0.13	0.09
3PC Vert.	1.28	1.68	0.68	1.31	1.89	0.60	1.34	1.55	1.47	2.05	0.95	1.40	1.73	1.35
1SIFT+1PC	0.32	0.68	0.15	0.61	0.79	0.11	0.44	0.37	0.37	1.12	0.23	0.44	0.62	0.47
1SIFT	**0.02**	**0.02**	**0.01**	**0.02**	**0.03**	**0.01**	**0.01**	**0.02**	**0.02**	**0.03**	**0.01**	**0.01**	**0.02**	**0.02**

from all inlier correspondences. Note that, when testing a solver aiming to find a particular plane (e.g., ground), we let GC-RANSAC decide which sample is good by immediately decomposing `H` to the pose and validating the essential matrix. Thus, the proposed solvers can be applied without a-priori knowing what types of planes are present in the scene.

The cumulative distribution functions of the rotation errors (in degrees) of the compared methods are shown in the left plot of Fig. 4. A method being accurate is interpreted as the curve being close to the top-left corner. Both of the proposed solvers are among the best performing ones. The cumulative distribution functions of the processing times and iteration numbers of the whole robust estimation procedure are shown in the right two plots of Fig. 4. The proposed 1SIFT ground solver leads to significantly faster robust estimation than the other algorithms. The average errors and processing times, for each scene, are reported in Table 2. It can be seen that the proposed 1SIFT ground solver is the fastest one on all scenes while being the second most accurate one.

4.4 Smart phone images

We captured two videos, at a resolution of 1920×1080 at 30Hz using an iPhone 6S, observing the ground or a vertical wall. The corresponding IMU data were captured at 100Hz. The frames and IMU data were synchronized based on their timestamps. The intrinsic camera parameters were obtained by using a Matlab toolbox. We applied the RealityCapture [34] software to acquire camera poses which we can use as ground truth when evaluating the solvers. For the sequence

Table 3: The average rotation error (in degrees), processing time (in seconds) and iteration numbers required for GC-RANSAC are reported.

Video 1 (ground)				Video 2 (wall)			
Solver	Error (°)	Time (s)	Iter. #	Solver	Error (°)	Time (s)	Iter. #
1SIFT	0.627	**0.571**	**286**	**1SIFT+1PC**	**5.364**	**0.219**	**233**
2PC Ground	**0.438**	0.842	973	3PC Vert.	5.391	0.420	853
5PC	1.875	2.966	5208	5PC	11.162	1.071	4104

showing the ground, we used every consecutive images. For the video of the wall, we used every 5th image. In total, the methods were tested on $1,247$ image pairs from the videos.

The results are reported in Table 3. For the video observing the ground, the proposed 1SIFT solver is marginally – by 0.2 degrees – less accurate than the 2PC solver of [6]. However, it leads to the most efficient robust estimation, being 0.3 seconds faster than the second fastest solver, i.e., 2PC Ground. For the video showing a wall, the proposed 1SIFT+1PC methods marginally leads to the most accurate solutions while being faster than the other algorithms. Furthermore, our solvers can be combined in a RANSAC loop to improve the performance.

5 Conclusions

We propose three new minimal solvers for estimating the egomotion of cameras with known vertical orientation and considering special target planes. The methods use orientation- and scale-covariant features, thus reducing the sample size at no cost when the features are obtained by the most commonly used detectors, e.g., SIFT. The proposed solvers are compared with state-of-the-art point-based approaches on both synthetic data and real images, showing comparable accuracy and significant improvements in computation times. Based on our experiments, the most promising method is the one which estimates the homography from a single correspondence if it originates from the ground plane. We believe that the proposed solvers will be useful for the community.

Acknowledgement

The research reported in this paper was supported by the OP VVV project Research Center for Informatics No. CZ.02.1.01/0.0/0.0/16_019/0000765 and by the project: Exploring the Mathematical Foundations of Artificial Intelligence 2018-1.2.1-NKP-00008.

References

1. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge university press (2003)

2. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24** (1981) 381–395
3. Nistér, D.: An efficient solution to the five-point relative pose problem. IEEE transactions on pattern analysis and machine intelligence **26** (2004) 756–770
4. Scaramuzza, D.: 1-point-ransac structure from motion for vehicle-mounted cameras by exploiting non-holonomic constraints. International journal of computer vision **95** (2011) 74–85
5. Choi, S., Kim, J.: Fast and reliable minimal relative pose estimation under planar motion. Image Vis. Comput. **69** (2018) 103–112
6. Saurer, O., Vasseur, P., Boutteau, R., Demonceaux, C., Pollefeys, M., Fraundorfer, F.: Homography based egomotion estimation with a common direction. IEEE transactions on pattern analysis and machine intelligence **39** (2017) 327–341
7. Fraundorfer, F., Tanskanen, P., Pollefeys, M.: A minimal case solution to the calibrated relative pose problem for the case of two known orientation angles. In: European Conference on Computer Vision, Springer (2010) 269–282
8. Saurer, O., Fraundorfer, F., Pollefeys, M.: Homography based visual odometry with known vertical direction and weak manhattan world assumption. In: Vicomor Workshop at IROS. Volume 2012. (2012)
9. Naroditsky, O., Zhou, X.S., Gallier, J., Roumeliotis, S.I., Daniilidis, K.: Two efficient solutions for visual odometry using directional correspondence. IEEE transactions on pattern analysis and machine intelligence **34** (2012) 818–824
10. Hee Lee, G., Pollefeys, M., Fraundorfer, F.: Relative pose estimation for a multi-camera system with known vertical direction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2014) 540–547
11. Ding, Y., Yang, J., Ponce, J., Kong, H.: An efficient solution to the homography-based relative pose problem with a common reference direction. In: The IEEE International Conference on Computer Vision (ICCV). (2019)
12. Ding, Y., Yang, J., Kong, H.: An efficient solution to the relative pose estimation with a common direction. In: 2020 IEEE International Conference on Robotics and Automation (ICRA), IEEE (2020) 11053–11059
13. Ding, Y., Yang, J., Ponce, J., Kong, H.: Minimal solutions to relative pose estimation from two views sharing a common direction with unknown focal length. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). (2020)
14. Ding, Y., Yang, J., Ponce, J., Kong, H.: Homography-based minimal-case relative pose estimation with known gravity direction. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
15. Guan, B., Vasseur, P., Demonceaux, C., Fraundorfer, F.: Visual odometry using a homography formulation with decoupled rotation and translation estimation using minimal solutions. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), IEEE (2018) 2320–2327
16. Barath, D., Toth, T., Hajder, L.: A minimal solution for two-view focal-length estimation using two affine correspondences. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 6003–6011
17. Barath, D., Hajder, L.: Efficient recovery of essential matrix from two affine correspondences. IEEE Transactions on Image Processing **27** (2018) 5328–5337
18. Guan, B., Zhao, J., Li, Z., Sun, F., Fraundorfer, F.: Minimal solutions for relative pose with a single affine correspondence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2020) 1929–1938

19. Hajder, L., Barath, D.: Relative planar motion for vehicle-mounted cameras from a single affine correspondence. IEEE International Conference on Robotics and Automation (ICRA) (2020)
20. Morel, J.M., Yu, G.: Asift: A new framework for fully affine invariant image comparison. SIAM journal on imaging sciences **2** (2009) 438–469
21. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A comparison of affine region detectors. International journal of computer vision **65** (2005) 43–72
22. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision **60** (2004) 91–110
23. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: European conference on computer vision, Springer (2006) 404–417
24. Barath, D., Kukelova, Z.: Homography from two orientation- and scale-covariant features. In: The IEEE International Conference on Computer Vision (ICCV). (2019)
25. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: Orb: An efficient alternative to sift or surf. In: 2011 International conference on computer vision, Ieee (2011) 2564–2571
26. Barath, D.: Five-point fundamental matrix estimation for uncalibrated cameras. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 235–243
27. Stewenius, H., Engels, C., Nistér, D.: Recent developments on direct relative orientation. ISPRS Journal of Photogrammetry and Remote Sensing **60** (2006) 284–294
28. Hartley, R., Li, H.: An efficient hidden variable approach to minimal-case camera motion estimation. IEEE transactions on pattern analysis and machine intelligence **34** (2012) 2303–2314
29. Kukelova, Z., Bujnak, M., Pajdla, T.: Polynomial eigenvalue solutions to minimal problems in computer vision. IEEE Transactions on Pattern Analysis and Machine Intelligence **34** (2012) 1381–1393
30. Sweeney, C., Flynn, J., Turk, M.: Solving for relative pose with a partially known rotation is a quadratic eigenvalue problem. 3DV **2** (2014) 5
31. Blanco-Claraco, J., Moreno-Dueñas, F., Jiménez, J.G.: The málaga urban dataset: High-rate stereo and lidar in a realistic urban scenario. I. J. Robotics Res. **33** (2014) 207–214
32. Barath, D., Matas, J.: Graph-cut ransac. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2018)
33. Ortin, D., Montiel, J.M.M.: Indoor robot motion based on monocular images. Robotica **19** (2001) 331–342
34. Capturing Reality: RealityCapture. http://www.capturingreality.com (2020)