Pose Estimation for Vehicle-mounted Cameras via Horizontal and Vertical Planes

Istvan Gergo Gal[1], Daniel Barath[2] and Levente Hajder[1]

Abstract— **We propose novel solvers for estimating the ego-motion of a calibrated camera mounted to a moving vehicle from a single affine correspondence via recovering special homographies. For the first, second and third classes of solvers, the sought plane is expected to be perpendicular to one of the camera axes. For the fourth class, the plane is orthogonal to the ground with unknown normal, *e.g.*, it is a building facade. All methods are solved via a linear system with a small coefficient matrix, thus, being extremely efficient. Both the minimal and over-determined cases can be solved by the proposed solvers. They are tested on synthetic data and on publicly available real-world datasets. The novel methods are more accurate or comparable to the traditional algorithms and are faster when included in state-of-the-art robust estimators. The source code is publicly available[1].**

I. Introduction

The estimation of plane-to-plane correspondences (*i.e.*, homographies) in an image pair is a fundamental problem for recovering the scene geometry. Recent state-of-the-art (SOTA) Structure from Motion [2], [3], [4] or Simultaneous Localization and Mapping [5], [6] algorithms combine epipolar geometry and homography estimation to be robust when the scene is close to being planar or the camera motion is rotation-only. In this paper, we use non-traditional input data (*i.e.*, affine correspondences) and focus on the case when the camera is mounted to a moving vehicle and there is a prior knowledge about the sought plane, for instance, it is the ground or a building facade.

An affine correspondence (AC) consists of a point pair and the related 2×2 local affine transformation, mapping the infinitesimally close vicinity of the point in the first image to the second one. Nowadays, a number of algorithms exist [7], [8], [9], [10], [11], [12], [13], [14], [15], [16] using ACs to estimate geometric entities, *e.g.*, homography, surface normal, epipolar geometry. These techniques are thoroughly discussed in the recent work of Barath et al. [17]. Affine features encode higher-order information about the scene geometry, thus the algorithms exploiting them solve the estimation problems from fewer correspondences than point-based methods. The reduced number of features is extremely important for randomized robust estimators, *e.g.*, RANSAC [18], where the processing time depends on the required number of points *exponentially*.

Recently, the attention is pointing towards autonomous driving, thus, it is becoming more and more important to design algorithms exploiting the properties of such a movement to provide results superior to general solutions. Considering that the cameras are moving on a plane, *e.g.*, they are mounted to a car, is a well-known approach for reducing the degrees-of-freedom and speeding up the robust estimation. Note that this assumption can be made valid if the vertical direction is known, *e.g.*, from an IMU sensor.

Ortin and Montiel [19] proved that, in case of planar motion, the epipolar geometry can be estimated from two point correspondences. This is also the motion model we assume in this paper. Since [19], several solvers have been proposed to estimate the motion from two correspondences [20], [21]. Scaramuzza [22] proposed a technique using a single point pair for a special camera setting assuming the special non-holonomic constraint to hold. The goal of this paper is to estimate camera pose for special vertical and horizontal planes when the camera motion is planar. The most related work to ours is the paper of Saurer et al. [23]. They estimate homographies from point correspondences by considering a prior knowledge about the normal of the sought plane, *e.g.*, it is orthogonal or parallel to the plane on which the vehicle, *i.e.*, typically a car or (quad)copter, moves. Contrary to [23], affine correspondences are exploited here as well.

Contributions. We propose solvers for estimating special homographies from *a single affine correspondence*. The addressed problem classes are visualized in Fig. 1. For the first type of solvers, the plane is assumed to be orthogonal to one of the camera axes. For the second one, the plane is vertical, *e.g.*, it is a facade of a building. The proposed methods solve linear systems, thus, being extremely fast, *i.e.*, 5–10 μs. The methods are tested on synthetic and on publicly available real-world datasets. They lead to accuracy comparable to the traditional algorithms while being significantly faster when included in SOTA robust estimators.

II. Problem Statement

Assume that we are given two calibrated cameras, with intrinsic camera matrices $\mathbf{K}$ and $\mathbf{K}'$, a planar object is observed. The world coordinate system is fixed to the first camera. The projection matrices are $\mathbf{P} = \mathbf{K}\left[\mathbf{I} \,|\, \mathbf{0}\right]$, $\mathbf{P}' =$

[1] Istvan Gergo Gal and Levente Hajder are with Department of Algorithms and their Applications, Eötvös Loránd University, Budapest, Hungary. L. Hajder is financed Thematic Excellence Programme TKP2020-NKA-06 National Challenges Subprogramme. I. G. Gal is supported by the project EFOP-3.6.3-VEKOP-16-2017-00001: Talent Management in Autonomous Vehicle Control Technologies. The projects are financed by the National Research, Development and Innovation Fund of Hungary, the Hungarian Government and co-financed by the European Social Fund.

[2] D. Barath is with VRG, Department of Cybernetics, Czech Technical University in Prague; MPLab, SZTAKI, Budapest; and Department of Computer Science, ETH Zurich. He is supported by the Ministry of Innovation and Technology NRDI Office within the framework of the Autonomous Systems National Laboratory Program and the OP VVV funded project CZ.02.1.01/0.0/0.0/16 019/0000765 “Research Center for Informatics”.

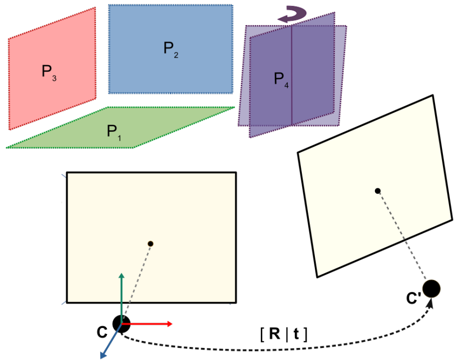


Fig. 1: Cameras $\mathbf{C}$ and $\mathbf{C}'$ are related by a rotation $\mathbf{R}$ around the vertical (yaw) axis and translation $\mathbf{t} = [t_x, 0, t_z]^{\mathrm{T}}$. Four different cases are considered, *i.e.*, when the points originate from a (1) horizontal P_1, (2) vertical frontal P_2, (3) vertical side P_3, and (4) general vertical plane P_4.

$\mathbf{K}' [\mathbf{R} \,|\, \mathbf{t}]$, where matrix $\mathbf{R}$ and vector $\mathbf{t}$ are, respectively, the 3D rotation and translation between the two views.

If there are corresponding points in the images, given by homogeneous coordinates as $\mathbf{u} = [x \quad y \quad 1]$ and $\mathbf{u}' = [x' \quad y' \quad 1]$, then the relationship w.r.t. the coordinates is linear. It is represented by a homography $\mathbf{H}$ as $\mathbf{u}' \sim \mathbf{H}\mathbf{u}$, where the operator $\sim$ denotes equality up to an usually unknown scale. In the case of calibrated cameras, the 2D coordinates can be normalized by the inverse of the intrinsic camera matrices. We use the normalized coordinates in the rest of this paper: $\mathbf{u} \leftarrow \mathbf{K}^{-1}\mathbf{u}$ and $\mathbf{u}' \leftarrow \mathbf{K}'^{-1}\mathbf{u}'$.

The homography parameters can be expressed via the relative pose [24], *i.e.*, rotation and translation, as follows:

$$\mathbf{H} \sim \mathbf{R} - \frac{1}{d}\mathbf{t}\mathbf{n}^{\mathrm{T}}, \tag{1}$$

where scalar d and vector $\mathbf{n}$ denote the distance of the observed plane from the first image and the normal of the plane, respectively. Operator $\sim$ denotes equality up to scale.

A. Planar motion

Assume that we are given a calibrated image pair with a common XZ plane (Y = 0), where axis Y is parallel to the vertical direction of the image planes. A trivial example for such a constraint is the camera setting of an autonomous car with a camera fixed to the moving vehicle and the Y axis of the camera being perpendicular to the ground plane. Note that this constraint can be straightforwardly made valid if the vertical direction is known, *e.g.*, from an IMU sensor. To estimate the camera motion, we first describe the parameterization of the problem.

Assuming planar motion, the rotation and translation are represented by three parameters: a 2D translation and the angle of rotation. Formally,

$$\mathbf{R} = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix}, \quad \mathbf{t} = \rho \begin{bmatrix} \cos\beta \\ 0 \\ \sin\beta \end{bmatrix}. \tag{2}$$

The translation is represented by length $\rho \in \mathbf{R}^{+}$ and $\beta \in [0, 2\pi)$. Angle $\alpha \in [0, 2\pi)$ is the rotation around axis Y.

B. Homography Estimation

In this paper, we exploit the relation between homographies and local affine frames. A homography $\mathbf{H}$ and an affinity $\mathbf{A}$ are represented by 3×3 by 2×2 matrices, respectively. We index their elements in a row-major order. Homography $\mathbf{H}$ represents the projective transformation between corresponding areas of planar surfaces in two images, while affine transformation $\mathbf{A}$ are defined as the first-order approximations of image-to-image transformations [10], including homographies. A homography is usually estimated from point correspondences in the images. If the point locations are denoted by vector $[x \quad y]^{\mathrm{T}}$ and $[x' \quad y']^{\mathrm{T}}$ in the first and second images, the relations between the coordinates [24] in the two views are as follows:

$$\begin{aligned} x'\left(h_7 x + h_8 y + h_9\right) &= h_1 x + h_2 y + h_3, \\ y'\left(h_7 x + h_8 y + h_9\right) &= h_4 x + h_5 y + h_6. \end{aligned} \tag{3}$$

Thus, each point correspondence (PC) adds two equations for the homography estimation.

Recently, Barath and Hajder proved [14] that the affine part of an affine correspondence gives four additional equations. They are as follows:

$$\begin{aligned} h_1 - \left(x' + a_1 x\right) h_7 - a_1 y h_8 - a_1 h_9 &= 0, \\ h_2 - \left(x' + a_2 y\right) h_8 - a_2 x h_7 - a_2 h_9 &= 0, \\ h_4 - \left(y' + a_3 x\right) h_7 - a_3 y h_8 - a_3 h_9 &= 0, \\ h_5 - \left(y' + a_4 y\right) h_8 - a_4 x h_7 - a_4 h_9 &= 0. \end{aligned} \tag{4}$$

In total, an affine correspondence (AC) provides six independent constraints. Consequently, one AC and two PCs are enough for estimating a general homography with eight degrees-of-freedom.

III. Proposed Methods

The following problem classes are considered: the estimation of (i) the ground plane, (ii,iii) special and (iv) general vertical planes. The objective is to recover the camera pose.

A. Ground Plane

The normal of the ground plane is $\mathbf{n} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^{\mathrm{T}}$. If planar motion is considered, Eq. 2 and normal $\mathbf{n}$ are substituted into Eq. 1. The homography becomes

$$\gamma\mathbf{H} = \begin{bmatrix} \cos\alpha & p & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & q & \cos\alpha \end{bmatrix}, \ \mathbf{H} \sim \begin{bmatrix} h_1 & h_2 & h_3 \\ 0 & h_5 & 0 \\ h_7 & h_8 & h_9 \end{bmatrix}, \tag{5}$$

where parameter γ denotes the unknown scale, $p = -\rho\cos\beta$ and $q = -\rho\sin\beta$, and the elements of $\mathbf{H}$ are $h_1 = h_9 = \cos\alpha$, $h_2 = p$, $h_3 = -h_7 = -\sin\alpha$, $h_8 = q$, $h_5 = 1$. Accordingly, three degrees-of-freedom (DoF) have to be estimated, *i.e.*, the unknown rotation angle α and a 2D translation represented by coordinates p and q.

If the relationship of the homography, point and affine parameters are considered (Eqs. 3 and 4), the estimation problem can be linearized as follows:

$$\mathbf{A}_{gd} \left[\cos\alpha \ \sin\alpha \ p \ q\right]^{\mathrm{T}} = \left[0 \ -y \ 0 \ 0 \ 0 \ -1\right]^{\mathrm{T}}, \tag{6}$$

where:

$$\mathbf{A}_{gd} = \begin{bmatrix} x - x' & -x'x - 1 & y & -x'y \\ -y' & -y'x & 0 & -y'y \\ 1 - a_1 & -x' - a_1 x & 0 & -a_1 y \\ -a_2 & -a_2 x & 1 & -x' - a_2 y \\ -a_3 & -y' - a_3 x & 0 & -a_3 y \\ -a_4 & -a_4 x & 0 & -y' - a_4 y \end{bmatrix}.$$

The point and affine parameters give six linear equations in the form of $\mathbf{A}_{gd}\mathbf{h}_{gd} = \mathbf{b}_{gd}$.

Optimal solver. The objective is to solve an inhomogeneous linear system with constraint $x_1^2 + x_2^2 = 1$, where $x_1 = \cos\alpha$ and $x_2 = \sin\alpha$ are the first two coordinates of vector $\mathbf{x}$. This algebraic problem can be optimally solved in the least squares sense via computing the intersections of two conics.

Rapid solver. The problem can be also solved by a homogeneous linear matrix equation $[\mathbf{A}_{gd}| - \mathbf{b}_{gd}]\,[\begin{array}{cc} \mathbf{x}^\mathrm{T} & 1 \end{array}]^\mathrm{T} = \mathbf{0}$. The null-vector of matrix $[\mathbf{A}_{gd}| - \mathbf{b}_{gd}]$ gives a suboptimal solution. Constraint $x_1^2 + x_2^2 = 1$ is made valid by dividing the obtained vector by its last coordinate. The angle is retrieved as $\alpha = \text{atan2}(x_2, x_1)$.

B. Special Vertical Planes

For urban scenes, it is quite frequent that planes of the buildings are parallel or perpendicular to the moving direction of the vehicle. In these cases, normals are $[1 \quad 0 \quad 0]^\mathrm{T}$ or $[0 \quad 0 \quad 1]^\mathrm{T}$. Although the homographies are not exactly the same as in Eq. 5, the problem is linear w.r.t. the same unknowns α, p and q. Therefore, the problem can be solved straightforwardly. The algebraic problems can be written as

$$\begin{aligned} \mathbf{A}_{v_1} \left[\cos\alpha, \sin\alpha, p, q\right]^\mathrm{T} &= \mathbf{b}_{v_1}, \\ \mathbf{A}_{v_2} \left[\cos\alpha, \sin\alpha, p, q\right]^\mathrm{T} &= \mathbf{b}_{v_2}, \end{aligned}$$

where $\mathbf{A}_{v_1}$ and $\mathbf{A}_{v_2}$ are the new coefficient matrices, and the right sides of the inhomogeneous problems are exactly the same as in Eq. 6, thus $\mathbf{b}_{v_1} = \mathbf{b}_{v_2} = \mathbf{b}_{gd}$. The coefficient matrices for the problem class are as follows:

$$\mathbf{A}_{v1} = \begin{bmatrix} x - x' & -x'x - 1 & -x & x'x \\ -y' & -y'x & 0 & y'x \\ 1 - a_1 & -x' - a_1 x & -1 & a_1 x + x' \\ -a_2 & -a_2 x & 0 & a_2 x \\ -a_3 & -y' - a_3 x & 0 & a_3 x + y' \\ -a_4 & -a_4 x & 0 & a_4 x \end{bmatrix},$$

and

$$\mathbf{A}_{v2} = \begin{bmatrix} x - x' & -x'x - 1 & -1 & x' \\ -y' & -y'x & 0 & y' \\ 1 - a_1 & -x' - a_1 x & 0 & a_1 \\ -a_2 & -a_2 x & 0 & a_2 \\ -a_3 & -y' - a_3 x & 0 & a_3 \\ -a_4 & -a_4 x & 0 & a_4 \end{bmatrix}.$$

C. General Vertical Planes

Assuming that the observed plane is vertical, with unknown orientation, is also an important case for autonomous driving. A general vertical wall has normal $\mathbf{n} = [n_x \quad 0 \quad n_z]^\mathrm{T}$. The surface normal itself can be represented by an angle δ as $\mathbf{n} = [\cos\delta \quad 0 \quad \sin\delta]^\mathrm{T}$. The implied homography is as follows:

$$\mathbf{H} = \begin{bmatrix} h_1 & 0 & h_3 \\ 0 & h_5 & 0 \\ h_7 & 0 & h_9 \end{bmatrix},$$

where $h_1 = (\cos\alpha - p\cos\delta)/\gamma$, $h_3 = (\sin\alpha - p\sin\delta)/\gamma$, $h_5 = 1/\gamma$, $h_7 = (-\sin\alpha - q\cos\delta)/\gamma$, and $h_9 = (\cos\alpha - q\sin\delta)/\gamma$. Therefore, the problem has five DoFs, *i.e.*, α, δ, γ, p and q. The six linear equations from Eqs. 3 and 4 form linear system:

$$\mathbf{A}_{vert}\mathbf{h} = \begin{bmatrix} 1 & 0 & 0 & -(x' + a_1 x) & -a_1 \\ 0 & 0 & 0 & -a_2 x & -a_2 \\ 0 & 0 & 0 & -(y' + a_3 x) & -a_3 \\ 0 & 0 & 1 & -a_4 x & -a_4 \\ x & 1 & 0 & -xx' & -x' \\ 0 & 0 & y & -xy' & -y' \end{bmatrix} \begin{bmatrix} h_1 \\ h_3 \\ h_5 \\ h_7 \\ h_9 \end{bmatrix} = \mathbf{0}. \qquad (7)$$

Solver. The elements of the homography matrix can be estimated by the null-matrix of $\mathbf{A}_{vert}$, and the scale-ambiguity, represented by variable γ, can be eliminated by scaling the homography matrix as $h_5 = 1$.

The remaining four parameters are retrieved from the scaled homography matrix. The elements are written as:

$$\begin{aligned} h_1 = \cos\alpha - p\cos\delta, \quad h_3 = \sin\alpha - p\sin\delta, \\ h_7 = -\sin\alpha - q\cos\delta, \quad h_9 = \cos\alpha - q\sin\delta. \end{aligned} \qquad (8)$$

From the 1st and 3rd equations, p and q are expressed as

$$p = \tfrac{\cos\alpha - h_1}{\cos\delta}, \quad q = -\tfrac{h_7 + \sin\alpha}{\cos\delta}. \qquad (9)$$

These are substituted back to the second and fourth equations. After elementary modifications, the following two equations are obtained: $h_3\cos\beta = h_1\sin\delta + \sin(\alpha - \delta)$, $h_9\cos\delta = h_7\sin\delta + \cos(\alpha - \delta)$. This can written by a matrix-vector product as:

$$\begin{bmatrix} h_9 & -h_7 \\ h_3 & -h_1 \end{bmatrix} \begin{bmatrix} \cos\delta \\ \sin\delta \end{bmatrix} = \begin{bmatrix} \cos(\alpha - \delta) \\ \sin(\alpha - \delta) \end{bmatrix},$$

that is a constrained matrix-vector equation: $\mathbf{B}\mathbf{v}_1 = \mathbf{v}_2$ *s.t.* $\mathbf{v}_1^\mathrm{T}\mathbf{v}_1 = \mathbf{v}_2^\mathrm{T}\mathbf{v}_2 = 1$. The SVD decomposition of matrix $\mathbf{B}$ is $\mathbf{B} = \mathbf{R}_1\text{diag}(\sigma_1^2, \sigma_2^2)\mathbf{R}_2$, where $\mathbf{R}_1$ and $\mathbf{R}_2$ are orthonormal matrices. The algebraic problem is as follows:

$$\mathbf{B}\mathbf{v}_1 = \mathbf{R}_1 \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix} \mathbf{R}_2\mathbf{v}_1 = \mathbf{v}_2.$$

Multiplying both sides by $\mathbf{R}_1^\mathrm{T}$ gives the formulas $\text{diag}(\sigma_1^2, \sigma_2^2)\mathbf{R}_2\mathbf{v}_1 = \mathbf{R}_1^\mathrm{T}\mathbf{v}_2$, where $\mathbf{v}_1^{'} = \mathbf{R}_2\mathbf{v}_1$ and $\mathbf{v}_2^{'} = \mathbf{R}_1^\mathrm{T}\mathbf{v}_2$. Finally, the formula to be solved is:

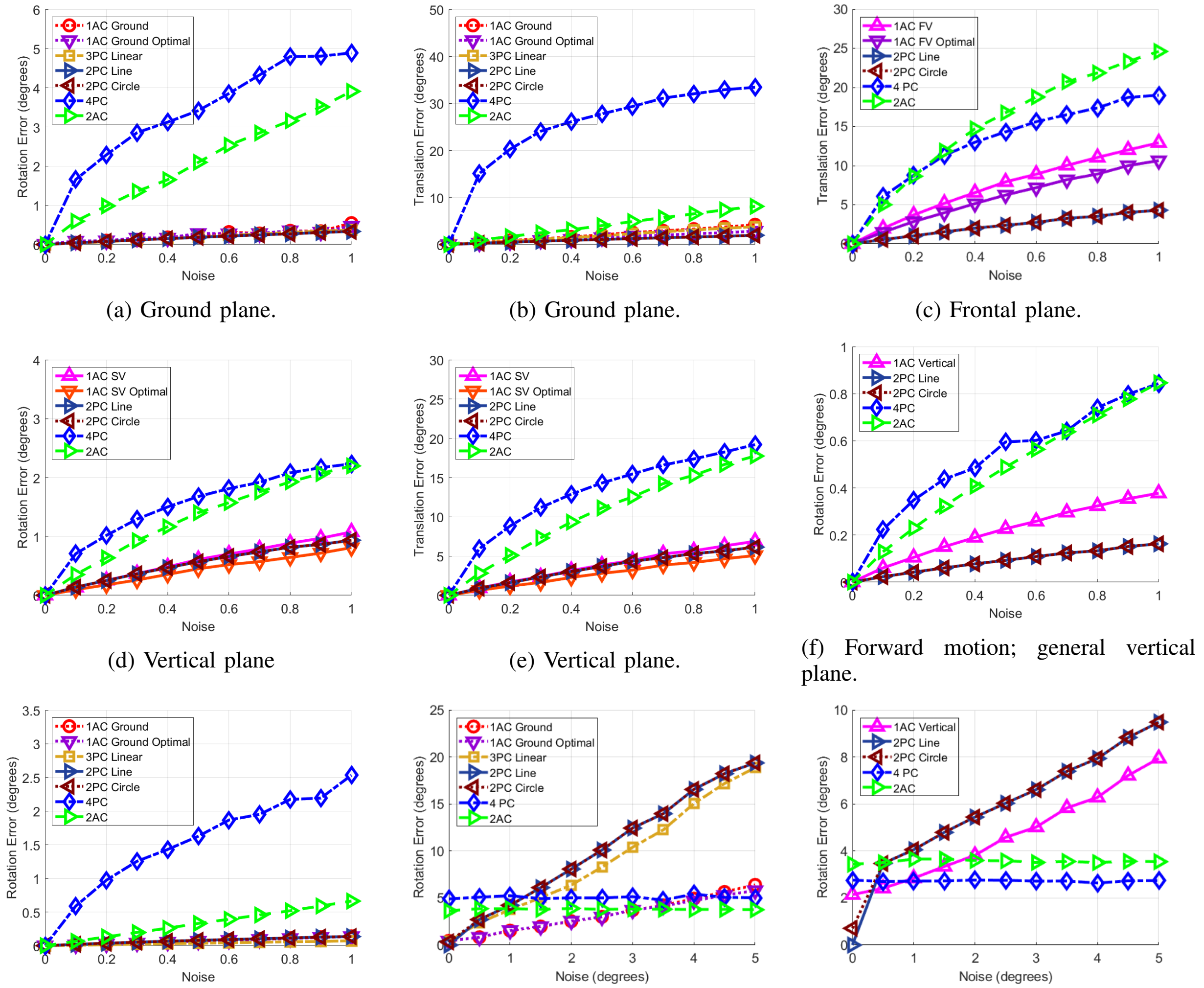
$$\begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix} \mathbf{v}_1^{'} = \mathbf{v}_2^{'}. \qquad (10)$$

Note that $\mathbf{v}_1^{'\mathrm{T}}\mathbf{v}_1^{'}$ = $\mathbf{v}_2^{'\mathrm{T}}\mathbf{v}_2^{'} = 1$ since a rotation does not change the length of a vector. This is a simple geometric problem: an origin-centered ellipse and an origin-centered circle with unit radius are on the left and right side of the equation, respectively. The intersections give four candidate solutions. The solution is straightforward and described in [21].

From the candidate solutions, the good one can be selected by the standard cheirality test [24] built on the fact that all 3D points, from which the pose is calculated, should be located in front of both cameras.

IV. Experimental Results

The proposed methods are tested on both synthetic and real-world image pairs from the Malaga dataset [25]. All the tested algorithms are our own implementations.



(a) Ground plane.

(b) Ground plane.

(c) Frontal plane.

(d) Vertical plane

(e) Vertical plane.

(f) Forward motion; general vertical plane.

(g) Forward motion; ground plane.

(h) Noisy gravity vector; ground plane.

(i) Noisy gravity; general vertical planes.

Fig. 2: Synthetic experiments. The compared methods are: the proposed ground-plane-based solvers (1AC Ground: rapid solver and 1AC Ground Optimal solver); the solvers assuming a frontal wall (1AC FV: front vertical rapid solver and 1AC FV optimal solver), wall on the side (1AC SV: side vertical solver rapid solver and 1AC SV optimal solver) or on a general vertical plane (1AC Vertical); the normalized DLT [24] algorithm (4PC) and the 2AC method [14] both estimating general homographies; 3PC Linear [21], 2PC Line [21], 2PC Circle [21] algorithms. In Figs. 2a - 2g three different noises are added to the affine correspondences. Noise in the point correspondence, affine scale, and rotation ranges from 0 to 1 pixel, percentage, and degree respectively.

A. *Synthetic Evaluation*

To evaluate the algorithms on synthetic data, we created a similar testing setup as [23]. The scene contains a ground plane, a wall in the front, a wall on the side and walls with general orientation. The distance of the planes from the 1st camera center is set to 10 unit distance (u.d.). The baseline of two cameras was set to 1 unit. The focal length was set to 1000 u.d. Each algorithm was evaluated under varying image noise. All the algorithms were tested under three types of motion, *i.e.* purely forward (along axis Z) and sideways movement (along axis X) and random planar motion.

To evaluate the accuracy, we compare the estimated relative poses. To measure the error in the relative rotation, we calculate the angular difference between the ground truth and estimated rotations as $\xi_R = \cos^{-1}((\mathrm{tr}(\mathbf{R}\dot{\mathbf{R}}^\mathrm{T})-1)/2)$, where $\dot{\mathbf{R}}$ is the ground truth and $\mathbf{R}$ is the estimated rotation. Since the translation is up scale, the error is the angular difference of the ground truth and estimated translations.

The proposed solvers are compared with the normalized DLT [24] (4PC), 2AC [14], 2PC Line and Circle [21], and 3PC Linear [21] solvers. Each test was repeated 10000 times. We considered two types of simulations. In the first one, the Y-axes of the cameras are parallel (Figs. 2a - 2g). In this case we increased the image noise from 0 to 1 pixel, while increasing the rotation and scale noise in the affine transformations, respectively, from 0 to 1 degrees and 0 to 1 percentages. In Figs. 2h - 2i, the sensitivity, of the proposed algorithms, is tested to the case when the camera motion is not entirely planar. For this purpose, the vertical direction of the second camera is rotated around axes X and Z with a small degree ranging from 0 to 5 while the other noise types are fixed to be 1 pixel (image noise), 1 degree (orientation) and 1 percentage (scale), respectively.

Ground plane. The proposed 1AC Ground Rapid and Optimal algorithms are compared with the general 4PC and 2AC, 3PC Linear, 2PC Line and Circle in Figs. 2a - 2b when the camera undergoes random planar motion, *i.e.*, it can move freely on its ground plane. The proposed methods have similarly low error as the top-performing solvers.

Special vertical planes. For scenes where the observed plane is in the front or on the side, the special cases of the 1AC Vertical algorithm are tested. Note that for the 3PC Linear algorithm scenes with vertical planes are degenerate cases and, thus, this solver was excluded from these experiments. Fig. 2c compares the special case with plane normal $[0 \quad 0 \quad 1]^{\mathrm{T}}$ under random planar motion. Noise is added to both the point coordinates and affine parameters. The proposed methods give better results than the general methods, but the SOTA 2PC-based methods are less sensitive to the noise. The rotation estimation shows similar behaviour but the plots are not included due to the lack of space. Figs. 2d - 2e compare the special case with plane normal $[1 \quad 0 \quad 0]^{\mathrm{T}}$ under random planar motion. The proposed 1AC Optimal solver leads to the most accurate results both in terms of rotation and translation errors.

Purely forward motion. Fig. 2f compares the solvers simulating purely forward motion in scenes with general vertical planes. The proposed method outperforms the general 2AC and 4PC methods, however, the algorithms of Choi et al. [21] are more accurate. For Fig. 2g, the tested scenes contain the ground plane. The proposed techniques lead to similarly low rotation errors as the top-performing ones.

Vertical direction. For Figs. 2h - 2i, we slightly invalidated the assumption that the cameras move on a plane by rotating them around their X and Z axes. As expected, the 4PC and 2AC methods are not affected by this noise due to assuming general motion. The proposed methods significantly outperform the 2PC Line and Circle solvers. The proposed 1AC Ground and Optimal methods are more accurate than the 4PC and 2AC general methods if the vertical noise is below approx. 3.0 degree. The proposed 1AC Vertical is more accurate than the 4PC and 2AC methods if the vertical noise is below approx. 1 degree. As shown in previous studies, e.g., [28], smartphones in 2009 such as Nokia N900 and iPhone 4 had a maximum gravity vector error of 1°. Nowadays, accelerometers used in cars and modern smartphones have noise levels around 0.06° (and expensive "good" accelerometers have $< 0.02^\circ$) [28].

B. Real-world experiments

To test the proposed techniques on real-world data, we chose the `Malaga` dataset [25]. This dataset was gathered entirely in urban scenarios with car-mounted sensors, including one high-resolution stereo camera and five laser scanners. We use the sequences of one high-resolution camera and every 10th frame from each sequence. The proposed method is applied to every consecutive image pair. The ground truth trajectories are composed using the GPS coordinates provided in the dataset. In total, 9 064 image pairs are used in the evaluation. To acquire affine correspondences [29] we use the VLFeat library [30], applying the Difference-of-Gaussians algorithm combined with the affine shape adaptation procedure as proposed in [31]. In our experiments, affine shape adaptation has only a small $\sim 10\%$ extra time demand over regular feature extraction. The correspondences are filtered by the standard SNN ratio test [29].

As a robust estimator, we choose Graph-Cut RANSAC [26] (GC-RANSAC). 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 are included in this part of the robust estimator. Also, it is observed that the considered special planes usually have lower inlier ratio, being localized in the image, compared to general ones. Therefore, we, instead of verifying the homography in the RANSAC loop, compose the essential matrix immediately from the recovered pose parameters and did not use the homography itself. For (b), we apply the eight-point relative pose solver to estimate the essential matrix from the larger-than-minimal set of inliers.

In the comparison, we use all methods which are added to the synthetic tests and, additionally, the 2PC Ground and 3PC Vertical [23] solvers, the 5PC [27] and the normalized 8PC [24] algorithms. We tested the proposed rapid and optimal solvers on real scenes and found that the difference in the accuracy is balanced by the robust estimator. We thus choose the rapid solver since it leads to no deterioration in the final accuracy, but speeds up the procedure.

The cumulative distribution functions of the rotation (in degrees), translation errors (in meters), and processing times (in seconds) are shown in Fig. 3. The error is calculated by decomposing the estimated essential/homography matrices to 3D rotation and translation. To calculate the translation errors in meters, we use the ground truth length from the dataset. A method being accurate is interpreted as the curve close to the top-left corner. Both of the proposed solvers (1AC Ground and 1AC Vertical) are among the most accurate methods. The processing times of the whole robust estimation procedure are shown in the right plot of Fig. 3. As it is expected, the proposed 1AC Ground method is *far the fastest* solver always returning its solution in at most $0.4-0.5$ seconds. In 80% of the cases, its processing time is less than 0.01 seconds. The proposed 1AC Vertical method leads to the second fastest robust estimation. The average rotation errors and processing times, for each scene, are reported in Table I. It can be seen that the proposed 1AC Ground solver is the *fastest on all scenes*. It leads also to the second most accurate rotations.

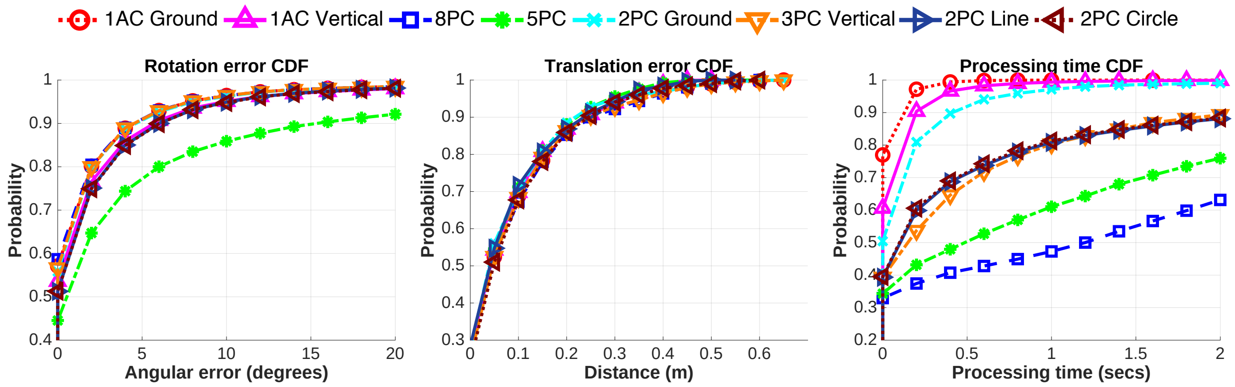


Fig. 3: The cumulative distribution functions of the rotation errors (in degrees), translation errors (in meters) and processing times (in seconds) on the 15 scenes (9 064 image pairs) of the Malaga dataset are shown. Being accurate or fast is interpreted by a curve close to the top-left corner. GC-RANSAC [26] is used as a robust estimator. The compared solvers are the proposed 1AC Ground, 1AC Vertical solvers; the eight (8PC)- [24] and five-point general methods (5PC) [27]; and the techniques from [23], 2PC Ground and 3PC Vertical and the 2PC-based algorithms of [21], 2PC Line and 2PC Circle.

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	avg.
Time (s)	1AC(G)	**0.12**	**0.11**	**0.04**	**0.08**	**0.03**	**0.10**	**0.18**	**0.12**	**0.10**	**0.31**	**0.06**	**0.13**	**0.07**	**0.07**	**0.08**	**0.11**
	1AC(V)	0.19	0.21	0.07	0.14	0.06	0.15	0.33	0.20	0.16	0.61	0.11	0.24	0.12	0.12	0.15	0.19
	8PC	4.57	3.82	0.96	2.55	0.79	4.56	5.71	3.36	2.49	4.54	1.72	3.64	2.63	1.92	1.86	3.01
	5PC	3.02	2.87	0.60	3.00	0.58	3.11	2.78	1.63	0.98	4.07	1.13	1.92	1.92	1.23	1.26	2.01
	2PC(G)	0.33	0.46	0.10	0.24	0.12	0.23	0.58	0.32	0.29	1.04	0.15	0.36	0.28	0.19	0.26	0.33
	3PC(V)	1.25	2.02	0.21	1.28	0.50	1.14	2.10	0.87	0.79	3.58	0.42	1.34	1.18	0.65	0.72	1.20
	2PC(L)	0.74	2.53	0.34	0.83	0.70	0.39	4.52	1.26	1.20	10.06	0.54	2.03	1.16	1.06	1.79	1.94
	2PC(C)	0.75	2.50	0.34	0.79	0.68	0.38	4.71	1.26	1.26	10.15	0.52	2.00	1.12	1.04	1.74	1.95
Ang. error (°)	1AC(G)	2.40	2.08	1.01	1.50	**3.64**	4.15	**4.63**	1.39	3.16	5.15	0.99	**2.52**	4.87	1.37	1.96	2.72
	1AC(V)	2.54	2.98	2.60	1.94	3.69	4.25	4.89	2.56	3.21	5.52	1.94	2.98	5.50	2.18	3.30	3.34
	8PC	2.42	**2.06**	**0.93**	**1.46**	**3.64**	4.17	4.79	**1.34**	**2.89**	5.18	**0.88**	2.53	**4.75**	**1.27**	1.95	**2.68**
	5PC	3.67	7.41	6.05	10.27	4.09	6.25	7.13	2.16	12.54	6.87	9.26	6.16	12.72	8.98	7.48	7.40
	2PC(G)	2.37	2.23	1.13	1.52	3.65	**4.14**	4.73	1.52	3.12	5.18	1.09	2.57	4.95	1.54	2.09	2.79
	3PC(V)	2.40	2.09	1.03	1.52	**3.64**	4.16	4.76	1.37	3.08	**5.09**	1.05	2.54	4.88	1.47	**1.93**	2.73
	2PC(L)	2.39	2.80	2.32	2.04	3.71	4.34	4.73	2.15	3.30	6.23	1.92	2.76	6.29	2.73	3.19	3.39
	2PC(C)	**2.36**	3.09	2.17	1.99	3.70	4.23	4.92	2.08	3.66	7.21	1.88	2.84	6.18	2.74	3.81	3.52

TABLE I: The average run-times (in seconds) and rotation errors (in degrees) of relative pose estimation on the 15 scenes (columns) of the Malaga dataset using different minimal solvers and Graph-Cut RANSAC as robust estimator [26]. The compared methods are the five-point solver of Stewenius et al. (5PC) [27], the normalized eight point solver (8PC) [24], two points on the ground (2PC(G)) and three points on a vertical plane (3PC(V)) solvers of Saurer et al. [23], the line-based (2PC(L)) and circle-based (2PC(C)) solvers of [21], and the proposed two affine-based solvers assuming points on the ground (1AC(G)) or on a vertical plane (1AC(V)). The corresponding cumulative distribution functions are shown in Fig. 3.

V. Conclusion and Future Work

We proposed minimal solvers for estimating the ego-motion of a calibrated camera mounted to a moving vehicle from a single affine correspondence assuming special planes to be observed. This problem is of fundamental importance for autonomous driving scenarios. The solvers are extremely efficient, *i.e.*, 5–10 μs in C++, as they are simplified to solving a linear system with a coefficient matrix of size 6×5. Also, due to using fewer correspondences than the state-of-the-art point-based solvers, the proposed methods significantly speed up the robust estimation procedure. The solver estimating the parameters of the ground plane lead to, on average, the second most accurate results on the approx. 9000 image pairs of the Malaga dataset.

The proposed methods can be inserted into real-time vision system of robots and (semi-)autonomous vehicles due to their speed. As human-made environments frequently contain horizontal and vertical planes, the methods can be used to detect such kind of planar objects with large surfaces.

References

[1] I. G. Gál. (2021) Matlab implementation of the proposed algorithms. [Online]. Available: https://github.com/Elenadar/Pose-Estimation-for-

Vehicle-mounted-Cameras-via-Horizontal-and-Vertical-Planes

[2] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: exploring photo collections in 3d," in *ACM Transactions on Graphics*, vol. 25, no. 3. ACM, 2006, pp. 835–846.

[3] N. Snavely, S. M. Seitz, and R, "Modeling the world from internet photo collections," *International Journal of Computer Vision*, vol. 80, no. 2, pp. 189–210, 2008.

[4] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4104–4113.

[5] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part i," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[6] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (slam): Part ii," *IEEE robotics & automation magazine*, vol. 13, no. 3, pp. 108–117, 2006.

[7] M. Perdoch, J. Matas, and O. Chum, "Epipolar geometry from two correspondences," in *International Conference on Pattern Recognition*, vol. 4, 2006, pp. 215–219.

[8] K. Köser, *Geometric estimation with local affine frames and free-form surfaces*, 2009, PhD. Thesis.

[9] J. Bentolila and J. M. Francos, "Conic epipolar constraints from affine correspondences," *Computer Vision and Image Understanding*, vol. 122, pp. 105–114, 2014.

[10] D. Barath, J. Molnár, and L. Hajder, "Optimal surface normal from affine transformation," in *International Conference on Computer Vision Theory and Applications*, vol. 2. SciTePress, 2015, pp. 305–316.

[11] C. Raposo and J. P. Barreto, "Theory and practice of structure-from-motion using affine correspondences," in *Computer Vision and Pattern Recognition*, 2016, pp. 5470–5478.

[12] C. Raposo and J. Barreto, "πmatch: Monocular vslam and piecewise planar reconstruction using fast plane correspondences," in *European Conference on Computer Vision*, 2016, pp. 380–395.

[13] D. Barath, T. Toth, and L. Hajder, "A minimal solution for two-view focal-length estimation using two affine correspondences," in *Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6003–6011.

[14] D. Barath and L. Hajder, "A theory of point-wise homography estimation," *Pattern Recognition Letters*, vol. 94, pp. 7–14, 2017.

[15] J. Pritts, Z. Kukelova, V. Larsson, and O. Chum, "Radially-distorted conjugate translations," *Conference on Computer Vision and Pattern Recognition*, pp. 1993–2001, 2018.

[16] D. Barath and L. Hajder, "Efficient recovery of essential matrix from two affine correspondences," *IEEE Transactions on Image Processing*, vol. 27, no. 11, pp. 5328–5337, 2018.

[17] D. Barath, M. Polic, W. Förstner, T. Sattler, T. Pajdla, and Z. Kukelova, "Making affine correspondences work in camera geometry computation," in *European Conference on Computer Vision*, 2020.

[18] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[19] D. Ortin and J. M. M. Montiel, "Indoor robot motion based on monocular images," *Robotica*, vol. 19, pp. 331–342, 2001.

[20] C. Chou and C. Wang, "2-point RANSAC for scene image matching under large viewpoint changes," in *International Conference on Robotics and Automation*, 2015, pp. 3646–3651.

[21] S. Choi and J. Kim, "Fast and reliable minimal relative pose estimation under planar motion," *Image Vision Computing*, vol. 69, pp. 103–112, 2018.

[22] 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.

[23] O. Saurer, P. Vasseur, R. Boutteau, C. Demonceaux, M. Pollefeys, and F. Fraundorfer, "Homography based egomotion estimation with a common direction," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 2, pp. 327–341, 2016.

[24] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

[25] J.-L. Blanco, F.-A. Moreno, and J. Gonzalez-Jimenez, "The málaga urban dataset: High-rate stereo and lidars in a realistic urban scenario," *International Journal of Robotics Research*, vol. 33, no. 2, pp. 207–214, 2014.

[26] D. Baráth and J. Matas, "Graph-cut RANSAC," *Conference on Computer Vision and Pattern Recognition*, pp. 6733–6741, 2018.

[27] H. Stewenius, C. Engels, and D. Nistér, "Recent developments on direct relative orientation," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 60, no. 4, pp. 284–294, 2006.

[28] 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 *European Conference on Computer Vision*. Springer, 2010, pp. 269–282.

[29] D. G. Lowe, "Object recognition from local scale-invariant features," in *International Conference on Computer Vision*, vol. 2. Ieee, 1999, pp. 1150–1157.

[30] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," https://www.vlfeat.org/.

[31] A. Baumberg, "Reliable feature matching across widely separated views," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2000, pp. 774–781.