International Journal of Computer Vision (2023) 131:2316–2332
https://doi.org/10.1007/s11263-023-01802-0

On Making SIFT Features Affine Covariant

Daniel Barath[1]

Received: 15 February 2022 / Accepted: 7 April 2023 / Published online: 1 June 2023

Abstract

An approach is proposed for recovering affine correspondences (ACs) from orientation- and scale-covariant, e.g., SIFT, features exploiting pre-estimated epipolar geometry. The method calculates the affine parameters consistent with the epipolar geometry from the point coordinates and the scales and rotations which the feature detector obtains. The proposed closed-form solver returns a single solution and is extremely fast, i.e., 0.5 μ seconds on average. Possible applications include estimating the homography from a single upgraded correspondence and, also, estimating the surface normal for each correspondence found in a pre-calibrated image pair (e.g., stereo rig). As the second contribution, we propose a minimal solver that estimates the relative pose of a vehicle-mounted camera from a single SIFT correspondence with the corresponding surface normal obtained from, e.g., upgraded ACs. The proposed algorithms are tested both on synthetic data and on a number of publicly available real-world datasets. Using the upgraded features and the proposed solvers leads to a significant speed-up in the homography, multi-homography and relative pose estimation problems with better or comparable accuracy to the state-of-the-art methods.

Keywords Covariant features · Affine correspondence · Relative pose · Homography · Epipolar geometry

1 Introduction

This paper addresses the problem of recovering fully affine-covariant features (Mikolajczyk et al. 2005) from orientation- and scale-covariant ones obtained by, for instance, SIFT (Lowe 1999) or SURF (Bay et al. 2006) detectors. This objective is achieved by exploiting the geometric constraints implied by the epipolar geometry of a pre-calibrated image pair. The proposed algorithm requires the epipolar geometry, i.e., either a fundamental **F** or an essential **E** matrix, and an orientation and scale-covariant feature as input. It returns the affine correspondence consistent with the epipolar geometry. Moreover, we propose a new solver that estimates the relative planar motion of a new image, given the calibrated pair, from a single SIFT correspondence and the corresponding normal.

Nowadays, a number of algorithms exist for estimating or approximating geometric models, e.g., homographies, using affine-covariant features. A technique, proposed by Perdoch et al. (2006), approximates the epipolar geometry from one or two affine correspondences by converting them to point pairs. Bentolila and Francos (2014) proposed a solution for estimating the fundamental matrix using three affine features. Raposo and Barreto (2016b, a) and Barath and Hajder (2018) showed that two correspondences are enough for estimating the relative pose of perspective cameras. Moreover, two feature pairs are enough for solving the semi-calibrated case, i.e., when the objective is to find the essential matrix and a common unknown focal length (Barath et al. 2017). Guan et al. (2021) proposed ways of estimating the generalized pose from affine correspondences. Also, homographies can be estimated from two affine correspondences as shown by Koser (2009), and, in the case of known epipolar geometry, from a single correspondence (Barath and Hajder 2017). There is a one-to-one relationship between local affine transformations and surface normals (Koser 2009; Barath et al. 2015a). Pritts et al. (2018) showed that the lens distortion parameters can be retrieved using affine features. Eichhardt and Barath (2020) demonstrated that a single correspondence equipped with monocular depth predictions is enough for estimating the two-view relative pose. The ways of using such solvers in practice are discussed in Barath et al. (2020) in depth.

Communicated by Federica Arrigoni.

✉ Daniel Barath
dbarath@ethz.ch

[1] Computer Vision and Geometry Group, ETH Zurich, Universitatstrasse 6., Zurich, Switzerland

Affine correspondences encode higher-order information about underlying the scene geometry. This is the reason why the previously mentioned algorithms solve geometric estimation problems (e.g., homographies and epipolar geometry) exploiting only a few correspondences—significantly fewer than what their point-based counterparts require. This, however, implies the major drawback of such techniques. Detectors for obtaining accurate affine correspondences, for example, Affine-SIFT (Morel and Yu 2009), Hessian-Affine or Harris-Affine (Mikolajczyk et al. 2005), MODS (Mishkin et al. 2015), HesAffNet (Mishkin et al. 2018), are slow compared to other widely used detectors. Thus, they are *not applicable* in time-sensitive applications, where real-time or close to real-time performance is required. In this paper, the objective is to bridge this problem by upgrading partially affine covariant features (e.g., SIFT providing the orientation and scale) to fully covariant ones exploiting the epipolar geometry.

In practice, *all* local features can be made orientation and scale covariant. Traditionally, feature detection involves three main steps: (scale-covariant) keypoint detection, orientation estimation, and descriptor extraction. Under such paradigm, keypoint detection is typically performed on the scale pyramid with help of a handcrafted response function, such as Hessian (Beaudet 1978; Mikolajczyk et al. 2005), Haris (Harris and Stephens 1988; Mikolajczyk et al. 2005), Difference of Gaussians (DoG, Morel and Yu (2009)), or learned ones like FAST (Rosten and Drummond 2006) or Key.Net (Barroso-Laguna et al. 2019). Keypoint detection gives a triplet (x, y, scale) which defines a square or circular patch. Then the patch orientation is estimated with help of handcrafted—dominant gradient orientation (Morel and Yu 2009), center of the mass (Rublee et al. 2011)—or learned methods providing both the feature scale and orientation (Yi et al. 2016; Mishkin et al. 2018; Lee et al. 2021). Finally, the patch is geometrically rectified and fed into a local patch descriptor, such as SIFT (Mishchuk et al. 2017), SOSNet (Tian et al. 2019), or other ones. Even though we mostly focus on SIFT features in this paper, where the orientation and scale is available without additional computations, the proposed algorithm can be applied to any features with orientations and scales estimated by one of the previously mentioned algorithms.

Exploiting feature orientation and scale for geometric model estimation is a known approach. In Mills (2018), the feature orientations are involved directly in the essential matrix estimation. In Barath (2017), the fundamental matrix is assumed to be a priori known and an algorithm is proposed for approximating a homography exploiting the rotations and scales of two SIFT correspondences. The approximative nature comes from the assumption that the scales along the axes are equal to the SIFT scale and the shear is zero. In general, these assumptions do not hold. The method of Barath (2018a) approximates the fundamental matrix by enforcing the geometric constraints of affine correspondences on the epipolar lines. Nevertheless, due to using the same affine model as in Barath (2017), the estimated epipolar geometry is solely an approximation. In Barath (2018b), a two-step procedure is proposed for estimating the epipolar geometry. First, a homography is obtained from three oriented features. Finally, the fundamental matrix is retrieved from the homography and two additional matches. Even though this technique considers the scales and shear unknowns, thus estimating the epipolar geometry instead of approximating it, the proposed decomposition of the affine matrix is not justified theoretically. Therefore, the geometric interpretation of the feature rotations is not provably valid. Barath (2018c) proposes a way of recovering full affine correspondences from the feature rotation, scale, and the fundamental matrix. Applying this method, a homography is estimated from a single correspondence in the case of known epipolar geometry. In Barath and Kukelova (2019), the authors propose a method to estimate the homography from two SIFT correspondences and provide a theoretically justifiable affine decomposition and general constraints on the homography.

Even though a number of solvers exist exploiting directly the orientation and scale from partially affine covariant features, many problems that are solvable from affine features remain unsolved from partially affine covariant ones. Such problems include, e.g., single-match homography, surface normal and relative pose estimation. Instead of proposing new solvers for such problems, we focus on upgrading features to be fully affine covariant so that they can be used within any existing affine-based solvers in a *light-weight* manner—due to not requiring an expensive image-based affine shape extraction procedure.

The contributions of the paper are: (i) we propose a technique for estimating affine correspondences from orientation- and scale-covariant features in the case of known epipolar geometry. The method is fast, i.e., $< 0.5\,\mu$s, and leads to a single solution. (ii) We propose a new solver that estimates the relative pose of a camera w.r.t. a pre-calibrated image pair from a single SIFT correspondence. It exploits affine correspondences and surface normals recovered by the proposed method between the pre-calibrated image pair. The solver assumes the cameras to move on a plane, e.g., by being rigidly mounted to a moving vehicle. Benefiting from the low number of correspondences required, robust homography and relative pose estimation, by e.g., GC-RANSAC (Barath and Matas 2018), is significantly faster than when using the traditional solvers while leading to more accurate results. Moreover, using the proposed technique for multi-homography estimation leads to an order-of-magnitude speed-up.

A preliminary version of the covariant feature upgrade algorithm was published in Barath (2018c). This paper extends and improves it by:

1. Using a theoretically justifiable affine decomposition model that connects the projection functions to the SIFT orientations and scales.
2. Simplifying the feature upgrade method so it returns only a single unique solution compared to the polynomial that returned up to two real solutions in Barath (2018c).
3. Proposing a new solver that exploits the upgraded features to estimate the relative pose of vehicle-mounted cameras from a single correspondence. Single-point solvers are highly relevant in practical applications due to reducing the robust estimation to a simple exhaustive search that has linear complexity in the number of input correspondences.
4. Providing a number of new experiments on homography, multi-homography, and relative pose estimation.

2 Theoretical Background

Affine correspondence $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{A})$ is a triplet, where $\mathbf{p}_1 = [u_1 \;\; v_1 \;\; 1]^\mathrm{T}$ and $\mathbf{p}_2 = [u_2 \;\; v_2 \;\; 1]^\mathrm{T}$ are a corresponding homogeneous point pair in two images and $\mathbf{A}$ is a 2×2 linear transformation which is called *local affine transformation*. Its elements in a row-major order are: a_1, a_2, a_3, and a_4. To define $\mathbf{A}$, we use the definition provided in Molnar and Chetverikov (2014) as it is given as the first-order Taylor-approximation of the 3D → 2D projection functions. For perspective cameras, the formula for $\mathbf{A}$ is the first-order approximation of the related *homography* matrix as:

$$\begin{aligned} a_1 &= \frac{\partial u_2}{\partial u_1} = \frac{h_1 - h_7 u_2}{s}, \quad a_2 = \frac{\partial u_2}{\partial v_1} = \frac{h_2 - h_8 u_2}{s}, \\ a_3 &= \frac{\partial v_2}{\partial u_1} = \frac{h_4 - h_7 v_2}{s}, \quad a_4 = \frac{\partial v_2}{\partial v_1} = \frac{h_5 - h_8 v_2}{s}, \end{aligned} \tag{1}$$

where u_i and v_i are the directions in the ith image ($i \in \{1, 2\}$) and $s = u_1 h_7 + v_1 h_8 + h_9$ is the projective depth. The elements of homography $\mathbf{H}$ in a row-major order are written as h_1, h_2,..., h_9.

The relationship of an affine correspondence and a homography is described by six linear equations. Since an affine correspondence involves a point pair, the well-known equations (from $\mathbf{H}\mathbf{p}_1 \sim \mathbf{p}_2$) relating the point coordinates hold (Hartley and Zisserman 2003). They are written as follows:

$$\begin{aligned} u_1 h_1 + v_1 h_2 + h_3 - u_1 u_2 h_7 - v_1 u_2 h_8 - u_2 h_9 &= 0, \\ u_1 h_4 + v_1 h_5 + h_6 - u_1 v_2 h_7 - v_1 v_2 h_8 - v_2 h_9 &= 0. \end{aligned} \tag{2}$$

After re-arranging Eq. (1), four additional linear constraints are obtained from $\mathbf{A}$ which are the following.

$$\begin{aligned} &h_1 - (u_2 + a_1 u_1) h_7 - a_1 v_1 h_8 - a_1 h_9 = 0, \\ &h_2 - (u_2 + a_2 v_1) h_8 - a_2 u_1 h_7 - a_2 h_9 = 0, \\ &h_4 - (v_2 + a_3 u_1) h_7 - a_3 v_1 h_8 - a_3 h_9 = 0, \\ &h_5 - (v_2 + a_4 v_1) h_8 - a_4 u_1 h_7 - a_4 h_9 = 0. \end{aligned} \tag{3}$$

Consequently, an affine correspondence provides six linear equations, in total, for the elements of the related homography matrix.

Fundamental matrix F relating the rigid background of two images is a 3×3 transformation matrix ensuring the so-called epipolar constraint $\mathbf{p}_2^\mathrm{T} \mathbf{F} \mathbf{p}_1 = 0$. Since its scale is arbitrary and $\det(\mathbf{F}) = 0$, matrix $\mathbf{F}$ has seven degrees-of-freedom (DoF).

The relationship of the epipolar geometry (either a fundamental or essential matrix) and affine correspondences are described in Barath et al. (2017) through the effect of $\mathbf{A}$ on the corresponding epipolar lines. Suppose that fundamental matrix $\mathbf{F}$, point pair $\mathbf{p}$, $\mathbf{p}'$, and the related affinity $\mathbf{A}$ are given. It can be proven that $\mathbf{A}$ transforms $\mathbf{v}$ to $\mathbf{v}'$, where $\mathbf{v}$ and $\mathbf{v}'$ are the directions of the epipolar lines ($\mathbf{v}, \mathbf{v}' \in \mathbb{R}^2$ *s.t.* $||\mathbf{v}||_2 = \left|\left|\mathbf{v}'\right|\right|_2 = 1$) in the first and second images (Bentolila and Francos 2014), respectively. It can be seen that transforming the infinitesimally close vicinity of $\mathbf{p}$ to that of $\mathbf{p}'$, $\mathbf{A}$ has to map the lines going through the points. Therefore, constraint $\mathbf{A}\mathbf{v} \parallel \mathbf{v}'$ holds.

As it is well-known from computer graphics (Turkowski 1990), formula $\mathbf{A}\mathbf{v} \parallel \mathbf{v}'$ can be reformulated as follows:

$$\mathbf{A}^{-\mathrm{T}} \mathbf{n} = \beta \mathbf{n}', \tag{4}$$

where $\mathbf{n}$ and $\mathbf{n}'$ are the normals of the epipolar lines ($\mathbf{n}, \mathbf{n}' \in \mathbb{R}^2$ *s.t.* $\mathbf{n} \perp \mathbf{v}$, $\mathbf{n}' \perp \mathbf{v}'$). Scalar β denotes the scale between the transformed and the original vectors if $||\mathbf{n}||_2 = \left|\left|\mathbf{n}'\right|\right|_2 = 1$. The normals are calculated as the first two coordinates of epipolar lines

$$\mathbf{l} = \mathbf{F}^\mathrm{T} \mathbf{p}' = [a \; b \; c]^\mathrm{T}, \quad \mathbf{l}' = \mathbf{F}\mathbf{p} = \left[a' \; b' \; c'\right]^\mathrm{T}. \tag{5}$$

Since the common scale of normals $\mathbf{n} = \mathbf{l}_{[1:2]} = [a \; b]^\mathrm{T}$ and $\mathbf{n}' = \mathbf{l}'_{[1:2]} = \left[a' \; b'\right]^\mathrm{T}$ comes from the fundamental matrix, Eq. (4) is modified as follows:

$$\mathbf{A}^{-\mathrm{T}} \mathbf{n} = -\mathbf{n}'. \tag{6}$$

Formulas (5) and (6) yield two equations which are linear in the parameters of the fundamental matrix as:

$$\begin{aligned} (u' + a_1 u) f_1 + a_1 v f_2 + a_1 f_3 + (v' + a_3 u) f_4 + \\ a_3 v f_5 + a_3 f_6 + f_7 = 0, \end{aligned} \tag{7}$$

$$\begin{aligned} a_2 u f_1 + (u' + a_2 v) f_2 + a_2 f_3 + a_4 u f_4 + \\ (v' + a_4 v) f_5 + a_4 f_6 + f_8 = 0. \end{aligned} \tag{8}$$

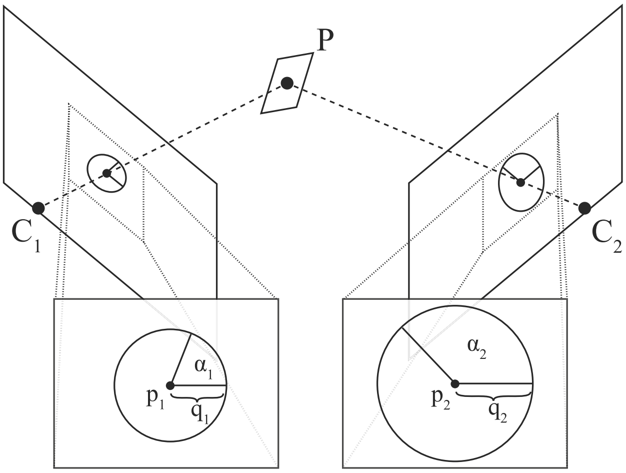


Fig. 1 Visualization of the orientation- and scale-covariant features. Point $\mathbf{P}$ and the surrounding patch projected into cameras $\mathbf{C}_1$ and $\mathbf{C}_2$. A window showing the projected points $\mathbf{p}_1 = [u_1 \ v_1 \ 1]^\mathrm{T}$ and $\mathbf{p}_2 = [u_2 \ v_2 \ 1]^\mathrm{T}$ are cut out and enlarged. The rotation of the feature in the ith image is α_i and the size is q_i ($i \in \{1, 2\}$). The scaling from the 1st to the 2nd image is calculated as $q = q_2/q_1$

where a_i is the ith element of $\mathbf{A}$ in row-major order, $i \in [1, 4]$. Points (u_1, v_1) and (u_2, v_2) are the points in, respectively, the first and second images.

To summarize this section, *the linear part* of a local affine transformation *gives two linear equations* for epipolar geometry estimation. A point correspondence yields a third one through the well-known epipolar constraint. Therefore, an affine correspondence leads to three linear constraints. As the fundamental matrix has seven Degrees-of-Freedom (DoF), three affine correspondences are enough for estimating $\mathbf{F}$ (Barath et al. 2020). Essential matrix $\mathbf{E}$ has five DoF and, thus, two affine correspondences are enough for the estimation (Barath and Hajder 2018).

3 Upgrade SIFT Features

In this section, we show how can affine correspondences be recovered from rotation- and scale-covariant features in the case of known epipolar geometry. Even though we will use SIFT as an alias for this kind of features, the derived formulas hold for every scale- and orientation-covariant ones. First, the affine transformation model is described in order to interpret the SIFT angles and scales. This model is substituted into the relationship of affine transformations and fundamental matrices. Finally, the obtained system is solved in closed-form to recover the unknown affine parameters.

3.1 Interpretation of SIFT Features

Reflecting the fact that we are given a scale $q_i \in \mathbb{R}^+$ and rotation $\alpha_i \in [0, 2\pi)$ independently in each image ($i \in \{1, 2\}$; see Fig. 1), the objective is to define affine correspondence $\mathbf{A}$ as a function of them. Such an interpretation was proposed in Barath and Kukelova (2019). However, the constraints in Barath and Kukelova (2019) have unnecessarily many unknowns, thus, complicating the estimation.

To understand the orientation and scale part of SIFT features, we exploit the definition of affine correspondences proposed in Barath et al. (2015a). In Barath et al. (2015a), $\mathbf{A}$ is defined as the multiplication of the Jacobians of the projection functions w.r.t. the image directions in the two images as follows:

$$\mathbf{A} = \mathbf{J}_2 \mathbf{J}_1^{-1}, \tag{9}$$

where $\mathbf{J}_1$ and $\mathbf{J}_2$ are the Jacobians of the 3D → 2D projection functions. Proof is in Appendix 1. For the ith Jacobian, we use the following decomposition:

$$\mathbf{J}_i = \mathbf{R}_i \mathbf{U}_i = \begin{bmatrix} \cos(\alpha_i) & -\sin(\alpha_i) \\ \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix} \begin{bmatrix} q_{u,i} & w_i \\ 0 & q_{v,i} \end{bmatrix}, \tag{10}$$

where angle α_i is the rotation in the ith image, $q_{u,i}$ and $q_{v,i}$ are the scales along axes u and v, and w_i is the shear ($i \in \{1, 2\}$). Plugging Eq. (10) into Eq. (9) leads to

$$\mathbf{A} = \mathbf{R}_2 \mathbf{U}_2 (\mathbf{R}_1 \mathbf{U}_1)^{-1} = \mathbf{R}_2 \mathbf{U}_2 \mathbf{U}_1^{-1} \mathbf{R}_1^\mathrm{T},$$

where $\mathbf{U}_1$ and $\mathbf{U}_2$ contain the unknown scales and shears in the two images. Since we are not interested in determining them separately, we can replace $\mathbf{U}_2 \mathbf{U}_1^{-1}$ by a single upper-triangular matrix $\mathbf{U} = \mathbf{U}_2 \mathbf{U}_1^{-1}$ simplifying the formula to

$$\begin{aligned} \mathbf{A} &= \mathbf{R}_2 \mathbf{U} \mathbf{R}_1^\mathrm{T} \\ &= \begin{bmatrix} \cos(\alpha_2) & -\sin(\alpha_2) \\ \sin(\alpha_2) & \cos(\alpha_2) \end{bmatrix} \begin{bmatrix} q_u & w \\ 0 & q_v \end{bmatrix} \begin{bmatrix} \cos(\alpha_1) & \sin(\alpha_1) \\ -\sin(\alpha_1) & \cos(\alpha_1) \end{bmatrix}. \end{aligned}$$

Angles α_1 and α_2 are known from the SIFT features. Let us use notation $c_i = \cos(\alpha_i)$ and $s_i = \sin(\alpha_i)$. The equations after the matrix multiplication become

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \\ &= \begin{bmatrix} c_2(c_1 q_u - s_1 w) + s_2 s_1 q_v & c_2(s_1 q_u + c_1 w) - s_2 c_1 q_v \\ s_2(c_1 q_u - s_1 w) - c_2 s_1 q_v & s_2(s_1 q_u + c_1 w) + c_2 c_1 q_v \end{bmatrix}. \end{aligned}$$

After simplifying the equations, we get the following linear system

$$\begin{aligned} a_1 &= c_2 c_1 q_u - c_2 s_1 w + s_2 s_1 q_v, \\ a_2 &= c_2 s_1 q_u + c_2 c_1 w - s_2 c_1 q_v, \\ a_3 &= s_2 c_1 q_u - s_2 s_1 w - c_2 s_1 q_v, \\ a_4 &= s_2 s_1 q_u + s_2 c_1 w + c_2 c_1 q_v, \end{aligned} \tag{11}$$

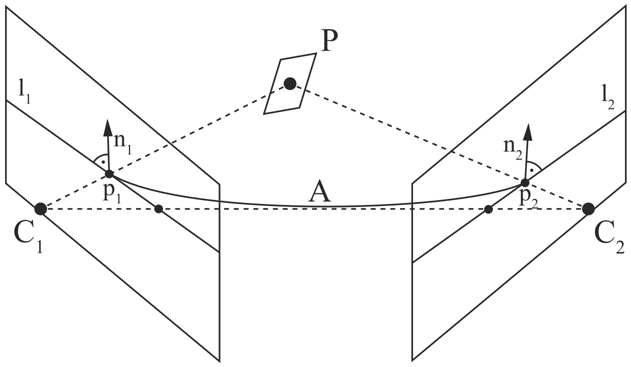


Fig. 2 The geometric interpretation of the relationship of a local affine transformations and the epipolar geometry (Eq. (6); proposed in Barath et al. (2017)). Given the projection $\mathbf{p}_i$ of $\mathbf{P}$ in the ith camera $\mathbf{C}_i$, $i \in \{1, 2\}$. The normal $\mathbf{n}_1$ of epipolar line $\mathbf{l}_1$ is mapped by affinity $\mathbf{A} \in \mathbb{R}^{2\times 2}$ into the normal $\mathbf{n}_2$ of epipolar line $\mathbf{l}_2$

where the unknowns are the affine parameters a_1, a_2, a_3, a_4, scales q_u, q_v and shear w.

In addition to the previously described constraints, we are given an extra one from the SIFT scale. It can be easily seen that the uniform scales of the SIFT features are proportional to the area of the underlying image region and, therefore, the scale change provides constraint

$$\det \mathbf{A} = \det\left(\mathbf{R}_2\mathbf{U}\mathbf{R}_1^{\mathrm{T}}\right) = \det \mathbf{U} = q_u q_v = \frac{q_2^2}{q_1^2}, \tag{12}$$

where q_1 and q_2 are the SIFT scales in the two images.

3.2 Epipolarly Consistent Affine Correspondences

Estimating the epipolar geometry as a preliminary step, either a fundamental or an essential matrix, is often done in computer vision applications. Also, when having a moving camera rig or a pre-built 3D map of the environment, we often are given both the intrinsic and extrinsic calibrations prior to the estimation. In the rest of this section, we consider fundamental matrix $\mathbf{F}$ to be known in order to exploit the relationship of epipolar geometry and affine correspondences proposed in Barath et al. (2017). Note that the formulas hold for essential matrix $\mathbf{E}$ as well.

For a local affine transformation $\mathbf{A}$ that is consistent with $\mathbf{F}$, formula Eq. (6) holds, where $\mathbf{n}_1$ and $\mathbf{n}_2$ are the normals of the epipolar lines, respectively, in the first and second images relating the regions around to the observed point locations (see Fig. 2). These normals are calculated as follows: $\mathbf{n}_1 = (\mathbf{F}^{\mathrm{T}}\mathbf{p}_2)_{(1:2)}$ and $\mathbf{n}_2 = (\mathbf{F}\mathbf{p}_1)_{(1:2)}$, where lower index $(1:2)$ selects the first two elements of the input vector (Figs. 3, 4). This relationship is formalized in Eqs. (7), (8). Assuming that $\mathbf{F}$ and point coordinates (u_1, v_1), (u_2, v_2) are known and the only unknowns are the affine parameters, Eqs. (7), (8) are reformulated as follows:

$$\begin{aligned} &(u_1 f_1 + v_1 f_2 + f_3)a_1 \\ +&(u_1 f_4 + v_1 f_5 + f_6)a_3 = -u_2 f_1 - v_2 f_4 - f_7, \\ &(u_1 f_1 + v_1 f_2 + f_3)a_2 \\ +&(u_1 f_4 + v_1 f_5 + f_6)a_4 = -u_2 f_2 - v_2 f_5 - f_8. \end{aligned} \tag{13}$$

These equations are linear in the affine components. Let us replace the constant parameters by new variables and, thus, introduce the following notation:

$$\begin{aligned} B &= u_1 f_1 + v_1 f_2 + f_3, \\ C &= u_1 f_4 + v_1 f_5 + f_6, \\ D &= -u_2 f_1 - v_2 f_4 - f_7, \\ E &= -u_2 f_2 - v_2 f_5 - f_8. \end{aligned}$$

Therefore, Eqs. (13) become

$$Ba_1 + Ca_3 = D, \quad Ba_2 + Ca_4 = E. \tag{14}$$

By substituting Eqs. (11) into Eqs. (14), the following formulas are obtained:

$$\begin{aligned} &B(c_2 c_1 q_u - c_2 s_1 w + s_2 s_1 q_v) \\ +&C(s_2 c_1 q_u - s_2 s_1 w - c_2 s_1 q_v) = D, \\ &B(c_2 s_1 q_u + c_2 c_1 w - s_2 c_1 q_v) \\ +&C(s_2 s_1 q_u + s_2 c_1 w + c_2 c_1 q_v) = E. \end{aligned} \tag{15}$$

Since the rotations and, therefore, their sinuses (s_1, s_2) and cosines (c_1, c_2), are considered to be known, Eqs. (15) are re-arranged as follows:

$$\begin{aligned} &(Bc_2c_1 + Cs_2c_1)q_u + (Bs_2s_1 - Cc_2s_1)q_v \\ &\qquad + (-Bc_2s_1 - Cs_2s_1)w = D, \\ &(Bc_2s_1 + Cs_2s_1)q_u + (Cc_2c_1 - Bs_2c_1)q_v + \\ &\qquad (Bc_2c_1 + Cs_2c_1)w = E. \end{aligned} \tag{16}$$

Let us introduce new variables encapsulating the constants as follows:

$$\begin{aligned} &G = Bc_2c_1 + Cs_2c_1, \quad H = Bs_2s_1 - Cc_2s_1, \\ &I = -Bc_2s_1 - Cs_2s_1, \quad J = Bc_2s_1 + Cs_2s_1, \\ &K = Cc_2c_1 - Bs_2c_1. \end{aligned}$$

Eqs. (16) then become

$$\begin{aligned} Gq_u + Hq_v + Iw &= D, \\ Jq_u + Kq_v + Gw &= E. \end{aligned} \tag{17}$$

From the first equation, we express w as follows:

$$w = \frac{D}{I} - \frac{G}{I}q_u - \frac{H}{I}q_v. \tag{18}$$

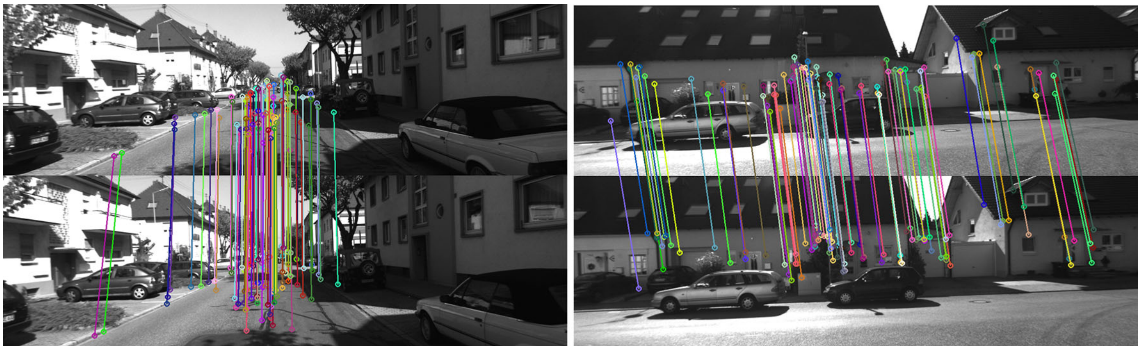
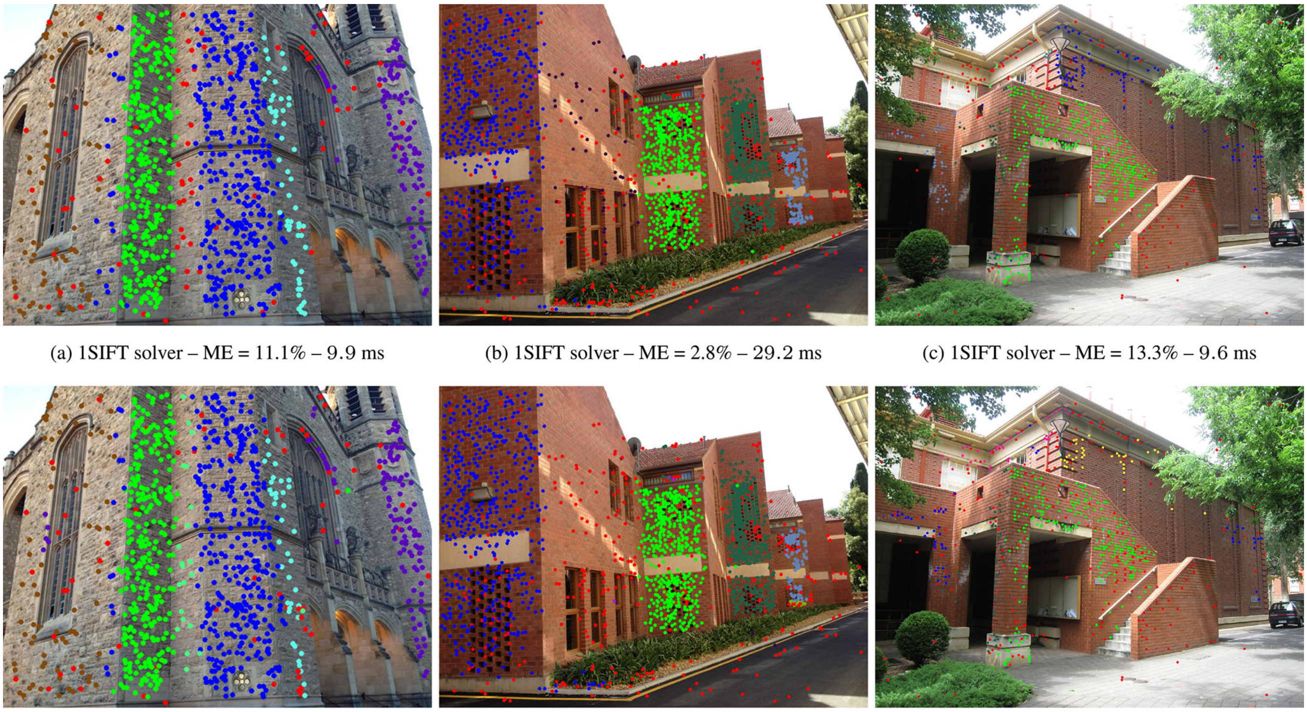


Fig. 3 Example image pairs and inliers of homographies estimated by the proposed 1SIFT solver from the KITTI dataset. Outliers are not visualized. Only 100 randomly selected inliers are drawn (Color figure online)



(a) 1SIFT solver – ME = 11.1% – 9.9 ms

(b) 1SIFT solver – ME = 2.8% – 29.2 ms

(c) 1SIFT solver – ME = 13.3% – 9.6 ms

(d) 4PC solver – ME = 11.1% – 54.8 ms

(e) 4PC solver – ME = 6.1% – 84.0 ms

(f) 4PC solver – ME = 28.0% – 34.1 ms

Fig. 4 Multi-homography fitting examples on the AdelaideRMF dataset using the proposed 1SIFT solver (top row) and the 4PC method (bottom) inside sequential GC-RANSAC. The misclassification error (ME) and run-time is written under the images. The point-to-homography assignment is denoted by color (red is outlier). Only the first image of the image pair is shown (Color figure online)

Let us notice that q_u and q_v are dependent due to Eq. (12) as $q_u = q_2^2/(q_1^2 q_v)$. By substituting this formula and Eq. (18) into the second equation of Eq. (17), the following quadratic polynomial equation is given:

$$\left(K - \frac{GH}{I}\right) q_v^2 + \left(\frac{GD}{I} - E\right) q_v + J\frac{q_2^2}{q_1^2} - \frac{G^2 q_2^2}{I q_1^2} = 0.$$

Let us notice that the coefficient of q_v^2 is always zero due to the trigonometric identities in expression $K - GH/I$. Therefore, the final formula for calculating q_v is simplified to

$$q_v = \frac{G^2 q_2^2 - IJq_2^2}{GDq_1^2 - EIq_1^2}.$$

All the other parameters can be straightforwardly calculated via $q_u = q_2^2/q_1^2 q_v$ and Eq. 18. Consequently, each SIFT correspondence can be upgraded to a fully affine covariant correspondence. Therefore, we recovered the local affine transformation from an orientation- and scale-covariant correspondence in the case of known epipolar geometry.

4 Planar Movement from a Single SIFT Feature with Normal

Assume that we are given a set of affine correspondences, e.g., recovered as proposed in the previous section, between a calibrated image pair and we aim at estimating the relative pose between the pair (I_1, I_2) and a new calibrated image I_3. Due to the stereo pair being calibrated, we can estimate the surface normal $\mathbf{n} \in \mathbb{R}^3$ from each affine correspondence independently as proposed in Barath et al. (2019). Let us form SIFT correspondences $(\mathbf{p}_1, \mathbf{p}_3, \alpha_1, \alpha_3, q_1, q_3, \mathbf{n})$ between the first image of the pair and the new one, where $\mathbf{p}_1, \mathbf{p}_3 \in \mathbb{R}^3$ are the points in their homogeneous form, $\alpha_1, \alpha_3 \in \mathbb{R}$ are the feature orientations and $q_1, q_3 \in \mathbb{R}$ are the sizes. The homography relating the features in the two images with normal $\mathbf{n}$ is

$$\mathbf{H} = \mathbf{R} - \frac{\mathbf{t}\mathbf{n}^{\mathrm{T}}}{d}, \tag{19}$$

where $\mathbf{R} \in \mathrm{SO}(3)$ is the relative rotation of the cameras, $\mathbf{t} \in \mathbb{R}^3$ is the relative translation, and $d \in \mathbb{R}$ is the plane intercept. Translation $\mathbf{t}$ is defined up-to-scale due to the perspective ambiguity (Hartley and Zisserman 2003). It is usual to let the translation absorb d such that we are given equation

$$\mathbf{H} = \mathbf{R} - \mathbf{t}\mathbf{n}^{\mathrm{T}}, \tag{20}$$

where, thus, the length of $\mathbf{t}$ has to be estimated.

Assuming that our cameras are mounted to a moving vehicle (i.e., planar movement), the expression can be further simplified. In this case, the rotation acts around the vertical axis and $t_y = 0$. The homography induced by normal $\mathbf{n}$ and a camera moving on a plane is as follows:

$$\begin{aligned}
\mathbf{H}_y &= \mathbf{R}_y - \mathbf{t}'\mathbf{n}^{\mathrm{T}} \\
&= \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} - \begin{bmatrix} n_x t_x & n_y t_x & n_z t_x \\ 0 & 0 & 0 \\ n_x t_z & n_y t_z & n_z t_z \end{bmatrix} \\
&= \begin{bmatrix} \cos\theta - n_x t_x & -n_y t_x & \sin\theta - n_z t_x \\ 0 & 1 & 0 \\ -\sin\theta - n_x t_z & -n_y t_z & \cos\theta - n_z t_z \end{bmatrix},
\end{aligned}$$

Let us replace $\cos\theta$ by α and $\sin\theta$ by β and introduce constraint $\alpha^2 + \beta^2 = 1$. The homography parameters become

$$\begin{aligned}
&h_1 = \alpha - n_x t_x, \quad h_2 = -n_y t_x, \; h_3 = \beta - n_z t_x, \\
&h_4 = 0, \qquad\qquad h_5 = 1, \qquad h_6 = 0, \\
&h_7 = -\beta - n_x t_z, \; h_8 = -n_y t_z, \; h_9 = \alpha - n_z t_z.
\end{aligned} \tag{21}$$

where the unknowns, in our case, are α, β, t_x, and t_z.

To estimate the homography, we are given the well-known two equations from Eq. (2) and two equations from Barath and Kukelova (2019) describing the relationship of SIFT features and homographies as follows:

$$\begin{aligned}
h_8 u_2 s_1 s_2 + h_7 u_2 s_2 c_1 - h_8 v_2 s_1 c_2 - h_7 v_2 c_1 c_2 + \\
-h_2 s_1 s_2 - h_1 s_2 c_1 + h_5 s_1 c_2 + h_4 c_1 c_2 = 0, \\
h_7^2 u_1^2 q_2 + 2h_7 h_8 u_1 v_1 q_2 + h_8^2 v_1^2 q_2 + h_5 h_7 u_2 q_1 + \\
-h_4 h_8 u_2 q_1 - h_2 h_7 v_2 q_1 + h_1 h_8 v_2 q_1 + 2h_7 h_9 u_1 q_2 + \\
2h_8 h_9 v_1 q_2 + h_2 h_4 q_1 - h_1 h_5 q_1 + h_9^2 q_2 = 0,
\end{aligned}$$

where the first equation is linear and the second one is quadratic in the homography parameters. Therefore, each SIFT correspondence leads to a quadratic and three linear equations.

We can use the three linear equations to solve for the unknown homography. Let us first plug Eq. (21) into the three linear equations as follows:

$$\begin{aligned}
&(u_1 - u_2)\alpha - (v_1 n_y + u_1 n_x + n_z)t_x \\
&\quad +(1 + u_1 u_2)\beta + (u_1 u_2 n_x + v_1 u_2 n_y + u_2 n_z)t_z = 0, \\
&\quad -v_2\alpha + u_1 v_2 \beta + (u_1 v_2 n_x + v_1 v_2 n_y + v_2 n_z)t_z = -v_1, \\
&\quad -s_2 c_1 \alpha + (v_2 c_1 c_2 - u_2 s_2 c_1)\beta + (n_x v_2 c_1 c_2 \\
&\quad +n_y v_2 s_1 c_2 - n_y u_2 s_1 s_2 - n_x u_2 s_2 c_1)t_z \\
&\quad +(n_y s_1 s_2 - n_x s_2 c_1)t_x = -s_1 c_2.
\end{aligned}$$

This is an under-determined inhomogeneous linear system of form $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{3\times 4}$, $\mathbf{x} = [\alpha, \beta, t_x, t_z]^{\mathrm{T}}$, and $\mathbf{b} = [0, -v_1, -s_1 c_2]^{\mathrm{T}}$. To solve it, we calculate the null-space of matrix $\mathbf{C} = [\mathbf{A} | -\mathbf{b}]$, add another unknown $s \in \mathbb{R}$ to $\mathbf{x}$ as $[\mathbf{x}^{\mathrm{T}}, s]^{\mathrm{T}}$, and introduce constraint $s = 1$. In this case, we are given five unknowns (i.e., α, β, t_x, t_y, s) and three constraints in matrix $\mathbf{C} \in \mathbb{R}^{3\times 5}$. Therefore, its right null-space will be two-dimensional. We can obtain it as the two eigenvectors corresponding to the two zero eigenvalues of matrix $\mathbf{C}^{\mathrm{T}}\mathbf{C}$. The solution vector $\mathbf{x}$ is calculated as

$$\mathbf{x} = \lambda_1 \mathbf{a} + \lambda_2 \mathbf{b},$$

where $\lambda_1, \lambda_2 \in \mathbb{R}$ are unknown scalars, $\mathbf{a}, \mathbf{b} \in \mathbb{R}^5$ are the null-vectors, and the elements of $\mathbf{x}$ are the solutions for the unknowns. Using constraint $s = \lambda_1 a_5 + \lambda_2 b_5 = 1$, we can express λ_1 as a function of λ_2 as:

$$\lambda_1 = \frac{1 - \lambda_2 b_5}{a_5}. \tag{22}$$

We can now use constraints $\alpha^2 + \beta^2 = 1$, where $\alpha = \lambda_1 a_1 + \lambda_2 b_1$ and $\beta = \lambda_1 a_2 + \lambda_2 b_2$. After plugging it in Eq. (22) and reformulating the expression, we get

$$\left(\left(b_1 - \frac{a_1 b_5}{a_5}\right)^2 + \left(b_2 - \frac{a_2 b_5}{a_5}\right)^2\right)\lambda_2^2 + 2\left(\frac{a_1}{a_5}\left(b_1 - \frac{b_5 a_1}{a_5}\right)\right.$$
$$\left. + \frac{a_2}{a_5}\left(b_2 - \frac{b_5 a_2}{a_5}\right)\right)\lambda_2 + \frac{a_1^2 + a_2^2}{a_5^2} - 1 = 0,$$

which is a quadratic polynomial in λ_2 with at most two real solutions. Parameter λ_1 has to be calculated for both solutions by substituting λ_2 into Eq. 22. Finally, the unknown parameters (i.e., the camera rotation around the vertical axis and the translation) are obtained as $\lambda_1 \mathbf{a} + \lambda_2 \mathbf{b}$. Both obtained solutions are equally valid and, thus, should be tested inside RANSAC. In summary, we can estimate the pose parameters from a single covariant feature correspondence (e.g., SIFT or ORB) when the camera undergo planar motion and we are given the surface normal. We call this solver 1SIFT + $\mathbf{n}$ in the rest of the paper.

Note that we did not use the quadratic constraint from Barath and Kukelova (2019) that could straightforwardly help in solving a more general problem, e.g., the $t_y \neq 0$ case when the vertical axes of the cameras are aligned by IMU readings.

5 Experimental Results

In this section, we compare the proposed algorithms both on synthetic and real-world data. The affine upgrade is shown to improve homography and multi-homography estimation on widely-used datasets. Moreover, the normal-based pose solver is shown to accelerate the pose estimation significantly while improving the accuracy too both with SIFT and SuperPoint (DeTone et al. 2018) features.

5.1 Affine Upgrade

For testing the accuracy of the affine correspondences obtained by the proposed method, we created a synthetic scene consisting of two cameras represented by their 3×4 projection matrices $\mathbf{P}_1$ and $\mathbf{P}_2$. They were located in random surface points of a 10-radius center-aligned sphere. A plane with random normal was generated in the origin and ten random points, lying on the plane, were projected into both cameras.

To get the ground truth affine transformations, we first calculated homography $\mathbf{H}$ by projecting four random points from the plane to the cameras and applying the normalized direct linear transformation (Hartley and Zisserman 2003). The local affine transformation regarding each correspondence was computed from the ground truth homography as its first-order Taylor-approximation by Eq. (1). Note that $\mathbf{H}$ could have been calculated directly from the plane parameters as well. However, using four points promised an indirect but geometrically interpretable way of noising the affine parameters: by adding zero-mean Gaussian-noise to the coordinates of the four projected points which implied $\mathbf{H}$. Finally, after having the full affine correspondence, $\mathbf{A}$ was decomposed to $\mathbf{R}_\alpha$, $\mathbf{R}_\beta$ and $\mathbf{U}$ by SVD decomposition to simulate the SIFT output. Since the decomposition is ambiguous, due to the two angles, β was set to a random value between 0 and 2π. Zero-mean Gaussian noise was added to the point coordinates and the affine transformations were noised in the previously described way.

The error of an estimated affinity is calculated as $\left|\left|\mathbf{I} - \widehat{\mathbf{A}}^{-1}\right|\right| \mathbf{A}_\text{F}$, where $\widehat{\mathbf{A}}$ is the estimated affine matrix, $\mathbf{A}$ is the ground truth one and norm $||.||_\text{F}$ is the Frobenious-norm.

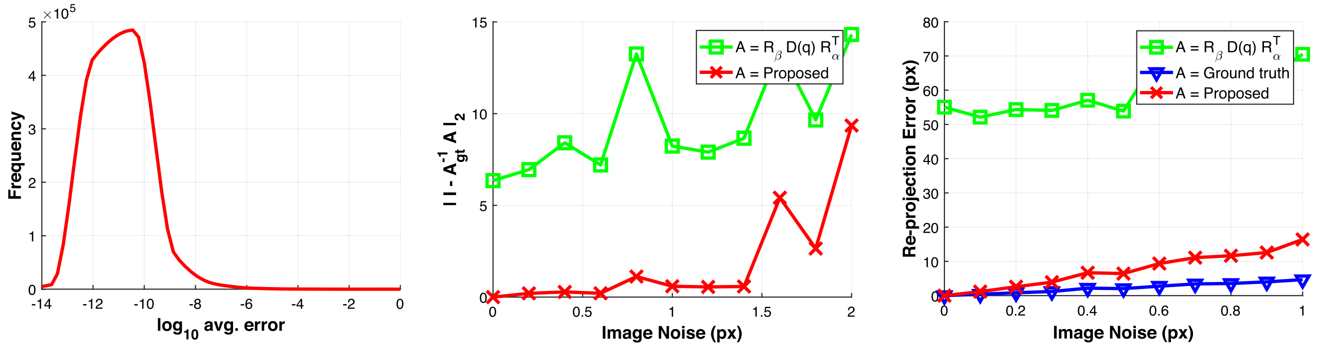
Figure 5a reports the numerical stability of the proposed method in the noise-free case. The frequencies (vertical axis), i.e., the number of occurrences in 100000 runs, are plotted as the function of the $\log_{10}$ average error (horizontal) computed from the estimated and ground truth affine transformations. It can be seen that the solver is numerically stable—all values are smaller than 10^{-6}.

Figure 5b reports the error of the estimated affinities plotted as the function of the noise σ added to the point coordinates. The affine transformations were estimated by the proposed method (red curve) and approximated as $\mathbf{A} \approx \mathbf{R}_\beta \mathbf{D} \mathbf{R}_{-\alpha}$ (green), where $\mathbf{R}_\theta$ is the 2D rotation matrix rotating by θ degrees and $\mathbf{D} = \text{diag}(q, q)$. As expected, approximating the affine frame by setting the shear to zero and assuming uniform scaling is inaccurate. Due to this rough approximation, the error is not zero even in the noise-free case. The proposed method leads to perfect results in the noise-free case and the error behaves reasonably as the noise increases.

5.2 Homography Estimation with Upgraded Features

In Barath and Hajder (2017), a method, called HAF, was published for estimating the homography from a single affine correspondence. The method requires the fundamental matrix and an affine correspondence to be known between the two images. These requirements fit well to the proposed algorithm, where we need to know the epipolar geometry and we return an affine correspondence for each SIFT-based one. This allows to estimate a homography for each correspondence even without without spending time on extracting fully affine covariant features.

Assuming that the underlying 3D point $\mathbf{P}$ lies on a continuous surface, HAF estimates homography $\mathbf{H}$ which the tangent plane of the surface at point $\mathbf{P}$ induces. The solution is obtained by first exploiting the fundamental matrix and reducing the number of unknowns in $\mathbf{H}$ to four. Then the relationship written in Eq. (1) is used to express the remaining homography parameters by the affine correspondences. The obtained inhomogeneous linear system consists of six equations for four unknowns. The problem to solve is $\mathbf{Cx} = \mathbf{b}$, where $\mathbf{x} = [h_7, \; h_8, \; h_9]^\text{T}$ is the vector of unknowns, i.e., the last row of $\mathbf{H}$, vector $\mathbf{b} = [f_4, \; f_5, \; -f_1, \; -f_2, \; -u_1 f_4 - v_1 f_5 - f_6, \; u_1 f_1 + v_1 f_2 - f_3]$ is the inhomogeneous part



(a)

(b)

(c)

Fig. 5 **a** *Stability study.* The frequencies (100000 runs; vertical axis) of $\log_{10}$ errors (horizontal) in the estimated affine transformations by the proposed method. **b** *Affine error.* The average errors in the affine transformation matrices estimated by the proposed algorithm (red curve) and the $\mathbf{A} \approx \mathbf{R}_\beta \mathbf{D}(q) \mathbf{R}_\alpha$ approximation (green) are plotted as a function of the image noise (in pixels; horizontal axis) added to the point coordinates. The error is calculated as $||\mathbf{I} - \widehat{\mathbf{A}}^{-1}\mathbf{A}||_\mathrm{F}$, where $\widehat{\mathbf{A}}$ is the estimated and $\mathbf{A}$ is the ground truth matrix. **c** *Homography error.* The average re-projection errors (in pixels) in the homographies estimated by the HAF algorithm from Barath and Hajder (2017) applied to the ground truth affine matrices (blue curve), the ones upgraded by the proposed methods (red), and to the approximated ones (green) (Color figure online)

and $\mathbf{C}$ is the coefficient matrix as follows:

$$\mathbf{C} = \begin{bmatrix} a_1 u_1 + u_2 - e'_u & a_1 v_1 & a_1 \\ a_2 u_1 & a_2 v_1 + u_2 - e'_u & a_2 \\ a_3 u_1 + v_2 - e'_v & a_3 v_1 & a_3 \\ a_4 u_1 & a_4 v_1 + v_2 - e'_v & a_4 \\ u_1 e'_u - u_1 u_2 & v_1 e'_u - v_1 u_2 & e'_u - u_2 \\ u_1 e'_v - u_1 v_2 & v_1 e'_v - v_1 v_2 & e'_v - v_2 \end{bmatrix},$$

where e'_u and e'_v are the coordinates of the epipole in the second image. The optimal solution in the least squares sense is $\mathbf{x} = \mathbf{C}^\dagger \mathbf{b}$, where $\mathbf{C}^\dagger = (\mathbf{C}^\mathrm{T}\mathbf{C})^{-1}\mathbf{C}^\mathrm{T}$ is the Moore-Penrose pseudo-inverse of $\mathbf{C}$.

According to the experiments in Barath and Hajder (2017), the method is often superior to the widely used solvers and makes the robust estimation significantly faster due to the low number of points needed. However, its drawback is the necessity of the affine features which are time consuming to obtain in real-world scenarios. By applying the algorithm proposed in this paper, it is possible to use the HAF method with SIFT features as input. Due to having real-time SIFT implementations, e.g., Sinha et al. (2006) and Acharya et al. (2018), the method is easy to be used in time-sensitive applications. In this section, we test the HAF method getting its input from the proposed algorithm both in our synthesized environment and on publicly available real-world datasets.

Synthetic experiments

For testing the accuracy of homography estimation, we used the same synthetic scene as for the previous experiments. For Fig. 5c, we calculated the essential matrix from the noisy point correspondences. Then, the homographies were estimated by HAF from affine correspondences recovered or approximated from the two rotations and the scale (similarly as in the previous section). Also, we ran HAF on the ground truth affine transformations with noisy essential matrix as an upper bound on what can be achieved by using perfect affine features.

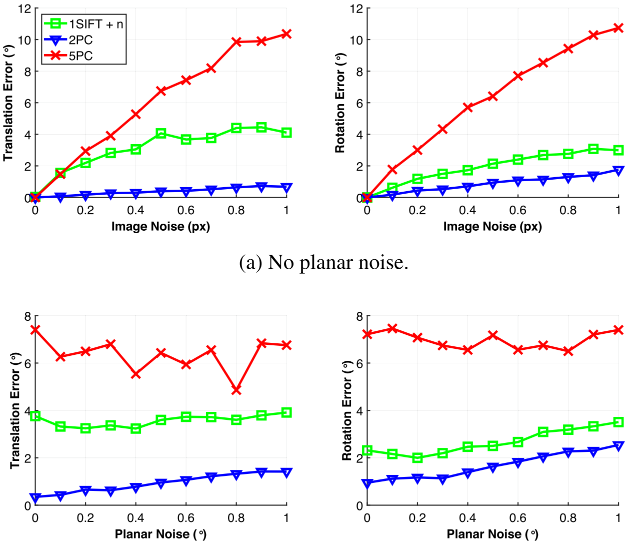
To measure the accuracy of $\mathbf{H}$, ten random points were projected into the cameras from the 3D plane inducing $\mathbf{H}$ and the average re-projection error was calculated (vertical axis; average of 1000 runs) and plotted as a function of the noise σ (horizontal axis). As expected, approximating the affine frame by setting the shear to zero and assuming uniform scaling along the axes leads to inaccurate homographies in Fig. 5c (green curve). Due to the approximation, the error is not zero even in the noise-free case. This is understandable since the applied HAF homography estimator uses a single affine correspondence. If this one AC comes from a rough approximation, the estimated homography will also be inaccurate. The proposed method leads to perfect results in the noise-free case and the error behaves reasonably as the noise increases (Fig. 6).

Real-world experiments

In order to test the proposed method on real-world data, we used the KITTI[1] and Malaga[2] datasets (see Figs. 3 and 7 for example image pairs). The KITTI odometry benchmark consists of 22 stereo sequences. Only 11 sequences (00–10) are provided with ground truth trajectories for training. We therefore used these 11 sequences to evaluate the compared solvers. The Malaga dataset was gathered entirely in urban scenarios with a car equipped with several sensors, including a high-resolution camera and five laser scanners. We used the 15 video sequences taken by the camera and every 10th image

[1] http://www.cvlibs.net/datasets/kitti/

[2] www.mrpt.org/MalagaUrbanDataset



(a) No planar noise.

(b) Fixed 0.5 px image noise.

Fig. 6 *Relative pose error* of the proposed 1SIFT + **n**, 2PC (Choi and Kim 2018), and 5PC (Stewenius et al. 2008) solvers plotted as a function of the image noise (top; in pixels) and planar noise (bottom; in degrees) (Color figure online)

from each sequence. In total, 23190 image pairs were used from the two datasets.

As a robust estimator, we chose Graph-Cut RANSAC (Barath and Matas 2018) with PROSAC (Chum and Matas 2005) sampling and inlier-outlier threshold set to 2 pixels. For the other parameters, we used the setting proposed by the authors.

We compare the proposed affine upgrade and HAF solver (1SIFT) with the normalized DLT algorithm (4PC), the three-point algorithm using the fundamental matrix (3PC, Barath and Hajder (2017)), the solver estimating the homography from three ORB features (3ORB, Barath (2018b)) and the 2SIFT solver (Barath and Kukelova 2019). We use the 4PC solver inside GC-RANSAC for non-minimal model estimation as suggested in Barath et al. (2020).

Given an image pair, the procedure to evaluate the estimators is as follows:

1. A fundamental matrix is estimated by GC-RANSAC using the seven-point algorithm as a minimal method, the normalized eight-point algorithm for least-squares fitting, the Sampson-distance as residual function, and a threshold set to 0.75 pixels. This threshold leads to accurate epipolar geometry on most of the tested scenes.
2. The homography is estimated by each tested solver using a 2 pixel threshold that was determined by tuning on the first sequence of KITTI to minimize the average error (tested thresholds: 0.5, 1, 2, and 3 pixels).
3. The homography is decomposed to rotation and translation. The angular rotation and translation errors are calculate w.r.t. the ground truth pose.

Table 1 reports the average and standard deviation of the rotation, translation (both in degrees), position (in meters) errors and, also, the run-time (in milliseconds). Even though none of the algorithms estimate the absolute scale, we cal-

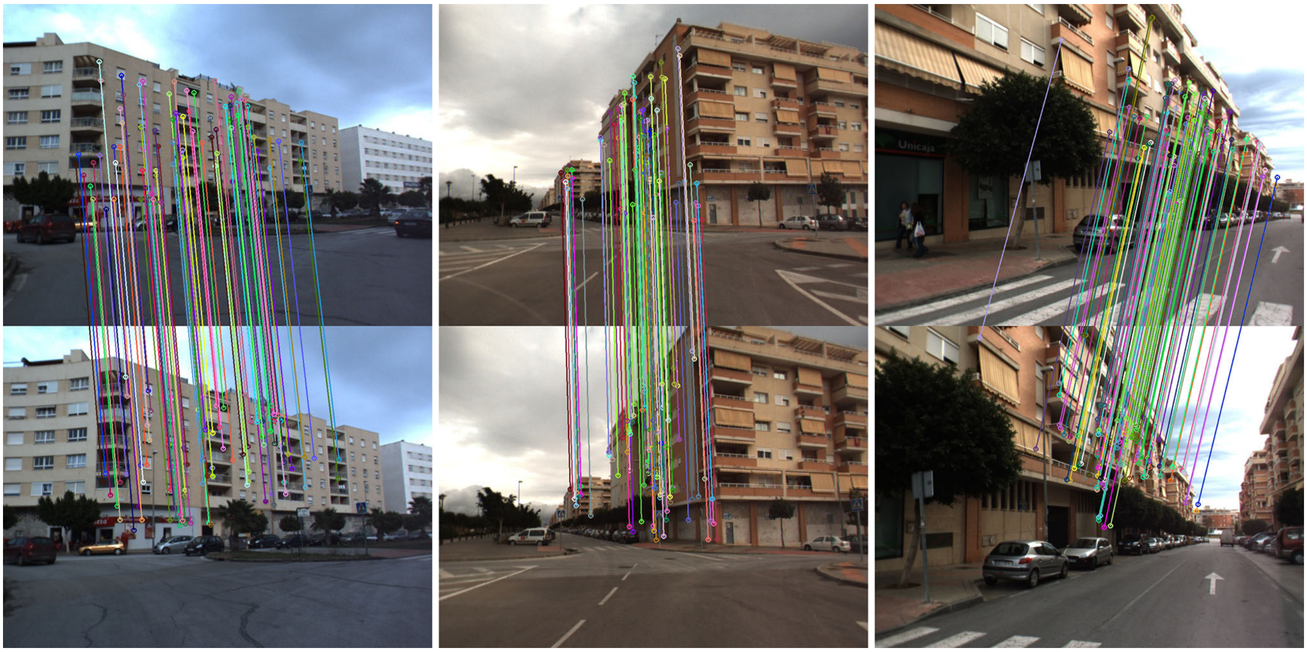


Fig. 7 Example image pairs and inliers of homographies estimated by the proposed 1SIFT solver from the Malaga dataset. Outliers are not visualized. Only 100 randomly selected inliers are drawn (Color figure online)

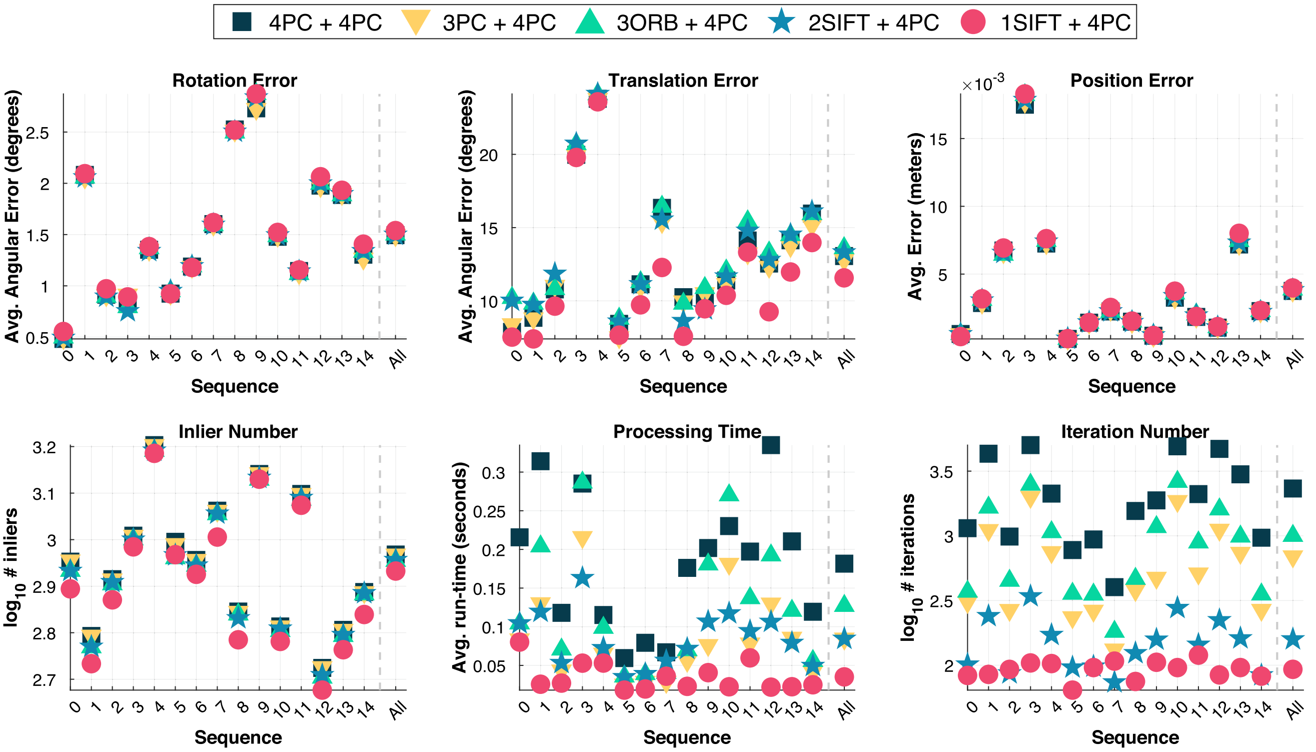


Fig. 8 The average rotation, translation (both in degrees), and position errors (in meters), the inlier and iteration numbers, and the run-time (in seconds) are shown on each scene of the Malaga dataset. GC-RANSAC (Barath and Matas 2018) is combined with different minimal solvers, i.e., normalized DLT (4PC), normalized DLT with known **F** (3PC), the 3ORB solver from Barath (2018b), the 2SIFT solver from Barath and Kukelova (2019), and the proposed one (1SIFT). The non-minimal solver is always the 4PC method. The average and std. over all scenes are shown in Table 1 (Color figure online)

Table 1 The average and standard deviation of the rotation, translation (both in degrees), position errors and the run-time (in milliseconds) on datasets KITTI (position in meters) and Malaga (in centimeters)

	Rotation (°)		Translation (°)		Position		Time (ms)	
	AVG	STD	AVG	STD	AVG	STD	AVG	STD
KITTI (15564)								
4PC	1.24	**1.65**	5.85	8.06	1.34	0.88	80.14	50.99
3PC	1.24	**1.65**	5.85	8.46	1.35	0.77	51.68	50.99
3ORB	1.26	**1.65**	6.25	8.85	1.36	0.87	112.37	115.66
2SIFT	1.25	**1.65**	6.72	8.87	1.51	**0.83**	82.80	**42.66**
1SIFT	**1.23**	**1.65**	**5.44**	**8.05**	**1.26**	0.87	**48.14**	44.41
Malaga (9049)								
4PC	**1.49**	2.10	13.06	11.72	**3.76**	2.67	181.58	135.25
3PC	**1.49**	2.22	12.79	11.34	3.77	2.68	84.65	81.23
3ORB	1.50	2.11	13.61	11.91	3.87	**2.64**	127.04	144.30
2SIFT	1.50	**2.09**	13.31	11.70	3.82	2.67	84.49	54.50
1SIFT	1.54	2.19	**11.57**	**10.70**	3.98	2.79	**35.18**	**30.20**

The number of image pairs are written in brackets. GC-RANSAC (Barath and Matas 2018) is combined with different minimal solvers, i.e., normalized DLT (4PC), normalized DLT with known **F** (3PC), the 3ORB solver from Barath (2018b), the 2SIFT solver from Barath and Kukelova (2019), and the proposed one (1SIFT). The non-minimal solver is always the 4PC method

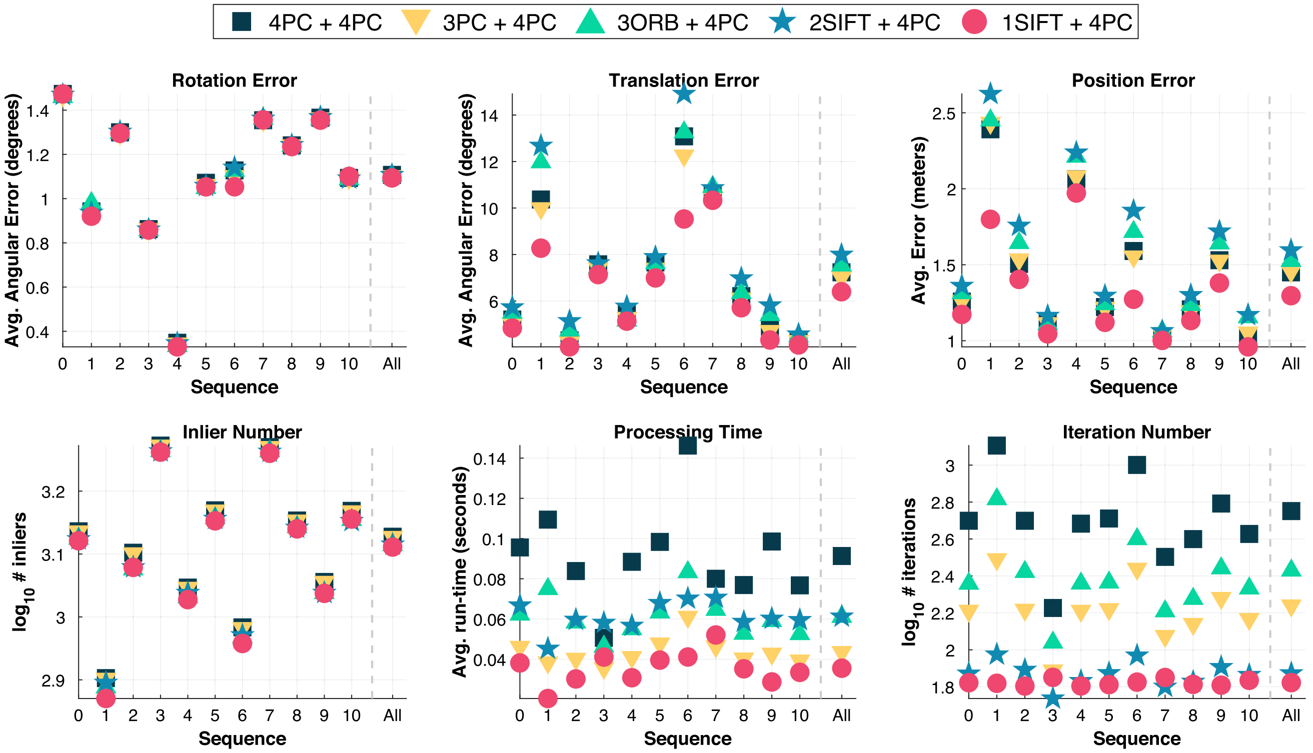


Fig. 9 The average rotation, translation (both in degrees), and position errors (in meters), the inlier and iteration numbers, and the run-time (in seconds) are shown on each scene of the KITTI dataset. GC-RANSAC (Barath and Matas 2018) is combined with different minimal solvers, i.e., normalized DLT (4PC), normalized DLT with known **F** (3PC), the 3ORB solver from Barath (2018b), the 2SIFT solver from Barath and Kukelova (2019), and the proposed one (1SIFT). The non-minimal solver is always the 4PC method. The average and std. over all scenes are shown in Table 1 (Color figure online)

culated the position error by using the ground truth scale with the estimated rotations and translations. The number of image pairs in each dataset is written in brackets. On the KITTI dataset, the proposed 1SIFT solver leads to the most accurate results while being, also, the fastest. On the Malaga dataset, it leads to the best translations while being the fastest and having comparable rotation and position estimates as other methods.

The rotation, translation and position errors, the inlier and iteration numbers, and the processing time on each scene of the tested datasets are shown in Figs. 8 and 9. The proposed 1SIFT solver has better or comparable accuracy to the most accurate methods. It is also the fastest on *all* but two sequences, i.e., the 3rd and 7th ones from the KITTI dataset, where it is the second fastest by a small margin.

5.3 Multi-homography Estimation with Upgraded Features

In this section, we apply the proposed solver to multi-homography estimation. For such problems, the outlier ratio tends to be extremely high for each homography to be found. Besides the mismatched points, the inliers of other homographies act as outliers when finding a particular one. Therefore, the size of the minimal sample required for the estimation is extremely important to reduce the combinatorics of the problem and allow finding the homographies efficiently.

To test the proposed solver, we downloaded the AdelaideRMF homography dataset from Wong et al. (2011). It consists of image pairs of resolution from 455 × 341 to 2592 × 1944 and manually annotated (assigned to a homography or to the outlier class) correspondences. Since the reference point sets do not contain rotations and scales, we detected and matched points applying the SIFT detector. We then found the closest match (in terms of average Euclidean distance) to each of the provided correspondences to find a ground truth set of matches. We removed a correspondence if the closest match from the manual annotation was more than 1 px far to remove gross outliers. We chose 1 px since it returns finds a pair for most of the ground truth matches while removing the outliers.

We run sequential GC-RANSAC which, in each iteration, finds the best homography and removes its inliers. We stop the estimation if the best homography has fewer than 8 inliers. We combine GC-RANSAC with the same minimal solvers as in the previous sections. The error of a method is measured

Table 2 Multi-homography estimation on the 18 pairs of AdelaideRMF dataset by sequential GC-RANSAC combined with minimal methods. Average and std. misclassification error (ME; percentages) and run-time (milliseconds) are reported. The tests are repeated 3 times in each scene

	ME (%)		Time (ms)	
	AVG	STD	AVG	STD
4PC	16.9	13.6	16.0	12.4
3PC	**12.9**	11.5	74.2	70.9
3ORB	26.7	14.6	26.7	14.6
2SIFT	17.8	13.9	19.1	21.4
1SIFT	13.3	**10.4**	**2.7**	**5.6**

Table 3 The rotation and translation errors (in degrees) and the run-time on 15564 image pairs from the KITTI dataset combining GC-RANSAC with different minimal solvers (1st column)

	Rotation (°)		Translation (°)		Time (ms)	
	AVG	MED	AVG	MED	AVG	MED
(I_k, I_{k+1})						
7PC	**1.2**	**0.5**	**2.0**	**1.2**	139.3	121.3
5PC	**1.2**	**0.5**	**2.0**	**1.2**	112.8	112.0
3PC Wall	**1.2**	**0.5**	**2.0**	**1.2**	119.2	106.3
2PC Ground	**1.2**	**0.5**	**2.0**	**1.2**	92.1	82.1
2PC Planar	**1.2**	**0.5**	**2.0**	**1.2**	100.0	81.4
1SIFT + n	**1.2**	**0.5**	2.1	**1.2**	**62.9**	**56.2**
(I_k, I_{k+3})						
7PC	7.3	1.8	9.0	1.5	126.3	124.2
5PC	4.3	**1.4**	4.3	**1.3**	365.2	417.4
3PC Wall	4.1	**1.4**	3.8	**1.3**	198.0	122.6
2PC Ground	4.1	**1.4**	3.9	**1.3**	52.7	37.7
2PC Planar	4.3	**1.4**	4.2	**1.3**	60.1	37.4
1SIFT + n	**3.9**	**1.4**	**3.4**	**1.3**	**16.1**	**14.7**

In the top part, the image pairs are (I_k, I_{k+1}) at frame k. In the bottom one, the image pairs are (I_k, I_{k+3}). The corresponding CDFs are shown in Fig. 10

by the misclassification errors as follows:

$$\mathrm{ME} = \frac{\#\ \text{misclassified points}}{\#\ \text{points}}.$$

Table 2 reports the average and std. of the misclassification error (in percentages) and run-time (in milliseconds) over 3 repetitions over all scenes. Using the proposed algorithm leads to the seconds most accurate results on average, with being only marginally, by 0.4%, less accurate than the best one. The proposed solver leads to the fastest model estimation (with 2.7 ms average time) and it is almost an order-of-magnitude faster than the second fastest method.

Example images are shown in Fig. 4. The misclassification error (ME) and run-time is written under the images. The point-to-homography assignment is denoted by color (red is outlier). Only the first image of the image pair is shown. The proposed method leads to better or similar accuracy than the widely used 4PC solver while being significantly faster.

5.4 Planar Relative Pose Solver

Synthetic experiments

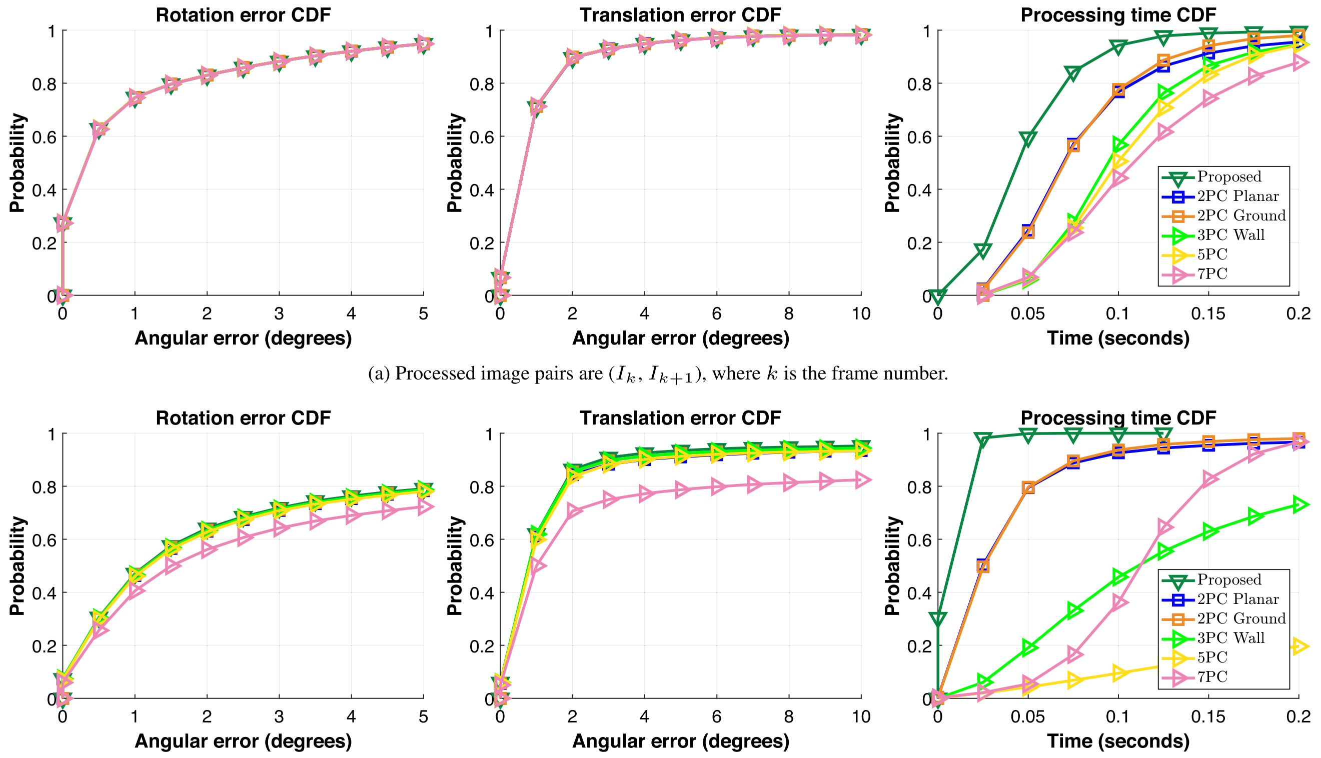
We compare the proposed 1SIFT + **n** solver to the 2PC (Choi and Kim, 2018) and 5PC (Stewenius et al., 2008) solvers in the same synthetic environment as what we used in the homography experiments. Figure 6a plots the rotation and translation errors as a function of the image noise. Since the normal are estimated from the noisy local affine transformations, they are also contaminated by noise. In this case, the cameras follow a perfect planar motion. The proposed solver is less accurate than the 2PC solver but more accurate than the 5PC in this case.

The errors w.r.t. the noise in the planarity assumption are plotted in Fig. 6b. We use a fixed 0.5 px image noise. To corrupt the planarity assumption, we rotated both the camera rotations and translations by a random rotation matrix with noise std. σ in degrees. The errors of both the 2PC and proposed 1SIFT + **n** solvers increase together with the planar noise. They both are more accurate than the 5PC method.

Real-world experiments

In this section, we demonstrate that the proposed SIFT-to-AC upgrade provides a *light-weight* way to equip the tentative correspondences with higher-order information about the underlying scene geometry. To do so, we assume that we are given a camera rig coming from, e.g., an actual rig or a pre-built map of the environment. For the sake of simplicity, we now assume that the rig consists of two cameras, I^1 and I^2, with pre-estimated relative rotation $\mathbf{R}_{\text{rig}} \in \mathrm{SO}(3)$ and translation $\mathbf{t}_{\text{rig}} \in \mathbb{R}^3$. The method can be straightforwardly extended to more cameras by selecting one as the center. The goal is to estimate the relative pose $(\mathbf{R}^1_{ij}, \mathbf{t}^1_{ij})$ of subsequent frames of the rig, denoted by the upper indices i and j $(i < j)$, assuming that the rig is centered on the first camera.

The first step is to detect and match SIFT correspondences both in I^1_i and I^2_i in the ith frame. Next, we apply the proposed SIFT-to-AC upgrade by using the known pose $\mathbf{R}_{\text{rig}}$ and $\mathbf{t}_{\text{rig}}$. Consequently, we obtain a set of affine correspondences for free in a *light-weight* manner, without an actual affine shape detector running, by using the known pose from the rig and the SIFT matches. Then, each of these ACs, the rotation $\mathbf{R}_{\text{rig}}$ and translation $\mathbf{t}_{\text{rig}}$ are fed into the optimal surface normal estimator proposed by Barath et al. (2015b) such that it estimates normal $\mathbf{n}$ for each correspondence in the coordinate system of I^1_i. Finally, correspondences are found between images I^1_i and I^1_j of the consecutive frames. We can now assign the estimated normals to the found correspondences and use them to estimate the relative pose between the frames of the rig using the proposed 1SIFT + **n** solver.



(a) Processed image pairs are (I_k, I_{k+1}), where k is the frame number.

(b) Processed image pairs are (I_k, I_{k+3}), where k is the frame number.

Fig. 10 The cumulative distribution functions of the rotation and translation errors (both in degrees), and the run-times (in seconds) of the proposed 1SIFT + **n** solver, 2PC from Choi and Kim (2018), 2PC ground and 3PC vertical plane solvers from Saurer et al. (2016), 5PC method from Stewenius et al. (2008) and 7PC solver from Hartley and Zisserman (2003) on the KITTI dataset. The tested minimal solvers were used in GC-RANSAC. The non-minimal solver is the 5PC algorithm. The pose is optimized by a final BA minimizing the pose error. Being accurate or fast is interpreted as a curve close to the top-left corner (Color figure online)

In this section, we test the previously described algorithm on the KITTI datasets, where we are given a moving stereo rig with known calibration. To obtain normals in the i-th frame, we first match SIFT features between the calibrated pair. We then use the method proposed in Sect. 3 to upgrade the features. From each estimated affine correspondence, we calculate the surface normal by the method proposed in Barath et al. (2015b). We then form SIFT matches between the first image of the rig in the i-th frame and the first image in the j-th frame. We now have surface normals for those correspondences, formed between frames t and $t + 1$, where the keypoint in the t-th frame was also matched to the second image of the rig. For all other matches, we assume that the normal is $[0, 1, 0]^T$, i.e., the point lies on a plane parallel to the ground plane.

We compare the proposed solver with the 2PC method from Choi and Kim (2018), the 2PC ground and 3PC vertical plane solvers from Saurer et al. (2016), the 5PC method from Stewenius et al. (2008) and the well-known 7PC solver from Hartley and Zisserman (2003) applied to estimate the relative pose between the first images of consecutive frames. All methods were used inside GC-RANSAC as minimal solver. The non-minimal solver is the 5PC algorithm. Finally, the pose is optimized by bundle adjustment (BA) minimizing the pose error.

The cumulative distribution functions (CDFs) of the rotation and translation errors (both in degrees) and the processing time (in seconds) are shown in Fig. 10. We tested the methods both on image pairs (I_k, I_{k+1}) and (I_k, I_{k+3}), where k is the frame index. In the (I_k, I_{k+1}) case, all tested solvers lead to similar accuracy both in terms of rotation and translation. The only difference is in the run-time, where the proposed method is significantly faster than the other competitors. In the (I_k, I_{k+3}) case, the accuracy drops slightly and the differences of the methods start becoming visible. The proposed solver leads to lower errors than the other ones. The processing time of almost all methods drops significantly compared to the (I_k, I_{k+1}) case due to the increased outlier ratio. The proposed one and the 2PC solvers, however, become faster due to the reduced number of point correspondences—they are not as sensitive to the outlier ratio as the other methods. The proposed 1SIFT + **n** solver is the fastest.

Table 4 The rotation and translation errors (in degrees) and the run-time on 15564 image pairs from the KITTI dataset combining GC-RANSAC with different minimal solvers (1st column) on SuperPoint features (DeTone et al. 2018) with feature scales and rotations estimated by Lee et al. (2021)

	Rotation (°)		Translation (°)		Time (ms)	
(I_k, I_{k+3})	AVG	MED	AVG	MED	AVG	MED
7PC	4.2	1.5	3.3	1.1	586.5	594.2
5PC	**3.8**	**1.4**	**2.2**	**1.0**	800.9	818.8
3PC Wall	4.0	1.5	2.5	**1.0**	296.9	173.7
2PC Ground	**3.8**	1.5	2.9	**1.0**	56.1	**41.6**
2PC Planar	4.0	1.5	3.2	**1.0**	69.0	41.8
1SIFT + n	**3.8**	**1.4**	2.3	**1.0**	**48.3**	44.3

The corresponding average and median values are shown in Table 3. In all cases, the proposed 1SIFT + **n** solver is the fastest. In the (I_k, I_{k+3}) case, it is also more accurate than the other algorithms.

To demonstrate that the proposed method is not limited to SIFT features, we applied SuperPoint (DeTone et al. 2018) followed by the scale and orientation extraction algorithm from Lee et al. (2021) to the images of the KITTI dataset. The results are reported in Table 4. The proposed 1SIFT + **n** solver leads to similar results as the 5PC algorithm, thus being amongst the most accurate methods, while being 20 times faster than 5PC.

6 Conclusion

An approach is proposed for recovering affine correspondences from orientation- and scale-covariant features obtained by, for instance, SIFT or SURF detectors. The method estimates the affine correspondence by enforcing the geometric constraints which the pre-estimated epipolar geometry implies. The solution is obtained in closed-form. Thus, the estimation is extremely fast, i.e., 0.5 µs, and leads to a single solution. Moreover, we propose a solver that estimates the planar motion from a single SIFT correspondence and the corresponding surface normal.

It is demonstrated both on synthetic and publicly available real-world datasets (containing approximately 25000 image pairs) that the proposed algorithm makes correspondence-wise homography estimation possible, thus, significantly speeding up the robust single and multi-homography estimation procedure. The proposed 1SIFT + **n** solver is designed for cases where the normal can be obtained prior to the estimation, e.g. in the multi-camera configuration or when we are given a known 3D map of the environment and the objective is to add a new image to the reconstruction.

Funding Open access funding provided by Swiss Federal Institute of Technology Zurich

A Proof the Affine Decomposition

We prove that decomposition $\mathbf{A} = \mathbf{J}_2\mathbf{J}_1^{-1}$, where $\mathbf{J}_i$ is the Jacobian of the projection function w.r.t. the directions in the ith image, is geometrically valid. Suppose that a three-dimensional point $\mathbf{P} = \begin{bmatrix} x \; y \; z \end{bmatrix}^\mathrm{T}$ lying on a continuous surface S is given. Its projection in the ith image is $\mathbf{p}_i = \begin{bmatrix} u_i \; v_i \end{bmatrix}^\mathrm{T}$. The projected coordinates, u_i and v_i, are determined by the projection functions $\mathbf{5}_u, \mathbf{5}_v : \mathbb{R}^3 \to \mathbb{R}$ as follows:

$$u_i = \mathbf{\Pi}_u^i(x, y, z), \;\; v_i = \mathbf{\Pi}_v^i(x, y, z),$$

where the coordinates of the surface point are written in parametric form as

$$x = \mathcal{X}(u, v), \;\; y = \mathcal{Y}(u, v), \;\; z = \mathcal{Z}(u, v).$$

It is well-known in differential geometry (Kreyszig 1968) that the basis of the tangent plane at point $\mathbf{P}$ is written by the partial derivatives of S w.r.t. the spatial coordinates. The surface normal $\mathbf{n}$ is expressed by the cross product of the tangent vectors $\mathbf{s}_u$ and $\mathbf{s}_v$ where

$$\mathbf{s}_u = \begin{bmatrix} \frac{\partial \mathcal{X}(u,v)}{\partial u} & \frac{\partial \mathcal{Y}(u,v)}{\partial u} & \frac{\partial \mathcal{Z}(u,v)}{\partial u} \end{bmatrix}^\mathrm{T},$$

and $\mathbf{s}_v$ is calculated similarly. Finally, $\mathbf{n} = \mathbf{s}_u \times \mathbf{s}_v$. Locally, around point $\mathbf{P}$, the surface can be approximated by the tangent plane, therefore, the neighboring points in the ith image are written as the first-order Taylor-series as follows:

$$\mathbf{p}_i \approx \mathbf{\Delta} \begin{bmatrix} \Pi_x(x, y, z) \\ \Pi_y(x, y, z) \end{bmatrix} + \begin{bmatrix} \frac{\partial \Pi_x^i(x,y,z)}{\partial u} & \frac{\partial \Pi_x^i(x,y,z)}{\partial v} \\ \frac{\partial \Pi_y^i(x,y,z)}{\partial u} & \frac{\partial \Pi_y^i(x,y,z)}{\partial v} \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix},$$

where $[\Delta v, \Delta u]^{\mathrm{T}}$ is the translation on surface S, and Δx, Δy are the coordinates of the implied translation added to $\mathbf{p}_i$. It can be seen that transformation $\mathbf{J}_i$ mapping the infinitely close vicinity around point $\mathbf{p}_i$ in the ith image is given as

$$\mathbf{J}_i = \begin{bmatrix} \frac{\partial \Pi_x^i(x,y,z)}{\partial u} & \frac{\partial \Pi_x^i(x,y,z)}{\partial v} \\ \frac{\partial \Pi_y^i(x,y,z)}{\partial u} & \frac{\partial \Pi_y^i(x,y,z)}{\partial v} \end{bmatrix},$$

thus

$$\left[\Delta x \ \Delta y\right]^{\mathrm{T}} \approx \mathbf{J}_i \left[\Delta u \ \Delta v\right]^{\mathrm{T}}.$$

The partial derivatives are reformulated using the chain rule. As an example, the first element it is as

$$\frac{\partial \Pi_x^i(x,y,z)}{\partial u} = \frac{\partial \Pi_x^i(x,y,z)}{\partial x} \frac{x}{\partial u} + \frac{\partial \Pi_x^i(x,y,z)}{\partial x} \frac{y}{\partial u} + \frac{\partial \Pi_x^i(x,y,z)}{\partial x} \frac{z}{\partial u} = \nabla(\mathbf{\Pi}_x^i)^{\mathrm{T}} \mathbf{s}_u,$$

where $\nabla \mathbf{\Pi}_x^i$ is the gradient vector of $\mathbf{\Pi}_x$ w.r.t. coordinates x, y and z. Similarly,

$$\frac{\partial \Pi_x^i}{\partial v} = \nabla(\mathbf{\Pi}_x^i)^{\mathrm{T}} \mathbf{s}_v,$$
$$\frac{\partial \Pi_y^i}{\partial u} = \nabla(\mathbf{\Pi}_y^i)^{\mathrm{T}} \mathbf{s}_u,$$
$$\frac{\partial \Pi_y^i}{\partial v} = \nabla(\mathbf{\Pi}_y^i)^{\mathrm{T}} \mathbf{s}_v.$$

Therefore, $\mathbf{J}_i$ can be written as

$$\mathbf{J}_i = \begin{bmatrix} \nabla(\mathbf{\Pi}_x^i)^{\mathrm{T}} \\ \nabla(\mathbf{\Pi}_y^i)^{\mathrm{T}} \end{bmatrix} \left[\mathbf{s}_u \ \mathbf{s}_v\right].$$

Local affine transformation $\mathbf{A}$ transforming the infinitely close vicinity of point $\mathbf{p}_1$ in the first image to that of $\mathbf{p}_2$ in the second one is as follows:

$$\begin{bmatrix} \Delta x_2 \\ \Delta y_2 \end{bmatrix} = \mathbf{J}_2 \mathbf{J}_1^{-1} \begin{bmatrix} \Delta x_1 \\ \Delta y_1 \end{bmatrix} = \mathbf{A} \begin{bmatrix} \Delta x_1 \\ \Delta y_1 \end{bmatrix}.$$

References

Acharya, K. A., Venkatesh Babu, R., & Vadhiyar, S. S. (2018). A real-time implementation of SIFT using GPU. *Journal of Real-Time Image Processing, 14*(2), 267–277.

Barath, D. (2017). P-HAF: Homography estimation using partial local affine frames. In *International conference on computer vision theory and applications*.

Barath, D. (2018a). Approximate epipolar geometry from six rotation invariant correspondences. In *International conference on computer vision theory and applications*.

Barath, D. (2018b). Five-point fundamental matrix estimation for uncalibrated cameras. In *Conference on computer vision and pattern recognition*.

Barath, D. (2018c). Recovering affine features from orientation-and scale-invariant ones. In *Asian conference on computer vision*.

Barath, D., & Hajder, L. (2017). A theory of point-wise homography estimation. *Pattern Recognition Letters, 94*, 7–14.

Barath, D., & Hajder, L. (2018). Efficient recovery of essential matrix from two affine correspondences. *IEEE Transactions on Image Processing, 27*(11), 5328–5337.

Barath, D., & Kukelova, Z. (2019). Homography from two orientation- and scale-covariant features. In: *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 1091–1099)

Barath, D., & Matas, J. (2018). Graph-Cut RANSAC. In *Conference on computer vision and pattern recognition*

Barath, D., Molnar, J., & Hajder, L. (2015a). Optimal surface normal from affine transformation. In *International joint conference on computer vision, imaging and computer graphics theory and applications*. SciTePress.

Barath, D., Molnar, J., & Hajder, L. (2015b). Optimal surface normal from affine transformation. In *International conference on computer vision theory and applications* (pp. 305–316). SciTePress.

Barath, D., Toth, T., & Hajder, L. (2017). A minimal solution for two-view focal-length estimation using two affine correspondences. In *Conference on computer vision and pattern recognition*.

Barath, D., Eichhardt, I., & Hajder, L. (2019). Optimal multi-view surface normal estimation using affine correspondences. *IEEE Transactions on Image Processing, 28*(7), 3301–3311.

Barath, D., Polic, M., Förstner, W., Sattler, T., Pajdla, T., & Kukelova, Z. (2020). Making affine correspondences work in camera geometry computation. In *European conference on computer vision* (pp. 723–740). Springer.

Barroso-Laguna, A., Riba, E., Ponsa, D., & Mikolajczyk, K. (2019). Key.Net: Keypoint detection by handcrafted and learned CNN filters. In *International conference on computer vision*.

Bay, H., Tuytelaars, T., & Van Gool, L. (2006). SURF: Speeded up robust features. In *European conference on computer vision*.

Beaudet, P. R. (1978). Rotationally invariant image operators. In *International joint conference on pattern recognition*.

Bentolila, J., & Francos, J. M. (2014). Conic epipolar constraints from affine correspondences. *Computer Vision and Image Understanding, 122*, 105–114.

Choi, S., & Kim, J. H. (2018). Fast and reliable minimal relative pose estimation under planar motion. *Image and Vision Computing, 69*, 103–112.

Chum, O., & Matas, J. (2005). Matching with PROSAC-progressive sample consensus. In *Computer vision and pattern recognition*.

DeTone, D., Malisiewicz, T., & Rabinovich, A. (2018). Superpoint: Self-supervised interest point detection and description. In *Conference on computer vision and pattern recognition workshops* (pp. 224–236).

Eichhardt, I., & Barath, D. (2020). Relative pose from deep learned depth and a single affine correspondence. In *European conference on computer vision* (pp. 627–644). Springer.

Guan, B., Zhao, J., Barath, D., & Fraundorfer F. (2021). Relative pose estimation for multi-camera systems from affine correspondences. In *International conference on computer vision*. IEEE.

Harris, C., & Stephens, M. (1988). A combined corner and edge detector. In *Alvey vision conference* (pp. 147–151).

Hartley, R., & Zisserman, A. (2003). *Multiple view geometry in computer vision*. Cambridge University Press.

Köser, K. (2009). Geometric estimation with local affine frames and free-form surfaces. Shaker.

Kreyszig, E. (1968). *Introduction to differential geometry and Riemannian geometry* (Vol. 16). University of Toronto Press.

Lee, J., Jeong, Y., & Cho, M. (2021). Self-supervised learning of image scale and orientation. In *British machine vision conference 2021*. BMVA Press.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *International conference on computer vision*.

Mikolajczyk, K., Tuytelaars, T., Schmid, C., et al. (2005). A comparison of affine region detectors. *International Journal of Computer Vision, 65*(1–2), 43–72.

Mills, S. (2018). Four-and seven-point relative camera pose from oriented features. In *International conference on 3D vision* (pp. 218–227). IEEE.

Mishchuk, A., Mishkin, D., Radenovic, F., & Matas J. (2017). Working hard to know your neighbor's margins: Local descriptor learning loss. In *Conference on neural information processing systems*.

Mishkin, D., Matas, J., & Perdoch, M. (2015). MODS: Fast and robust method for two-view matching. *Computer Vision and Image Understanding, 141*, 81–93.

Mishkin, D., Radenovic, F., & Matas, J. (2018). Repeatability is not enough: Learning affine regions via discriminability. In *Proceedings of the European conference on computer vision (ECCV)* (pp 284–300).

Molnár, J., & Chetverikov, D. (2014). Quadratic transformation for planar mapping of implicit surfaces. *Journal of Mathematical Imaging and Vision, 48*, 176–184.

Morel, J. M., & Yu, G. (2009). ASIFT: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences, 2*(2), 438–469.

Perdoch, M., Matas, J., & Chum, O. (2006). Epipolar geometry from two correspondences. In *International conference on pattern recognition*.

Pritts, J., Kukelova, Z., Larsson, V., & Chum, O. (2018). Radially-distorted conjugate translations. In *Conference on computer vision and pattern recognition*.

Raposo, C., & Barreto, J. P. (2016a). πmatch: Monocular vSLAM and piecewise planar reconstruction using fast plane correspondences. In *European conference on computer vision* (pp. 380–395). Springer.

Raposo, C., Barreto, J. P. (2016b). Theory and practice of structure-from-motion using affine correspondences. In *Computer vision and pattern recognition*.

Rosten, E., & Drummond, T. (2006). Machine learning for high-speed corner detection. In *European conference on computer vision* (pp. 430–443). Springer-Verlag, Berlin, Heidelberg, ECCV'06. https://doi.org/10.1007/11744023_34.

Rublee, E., Rabaud. V., Konolidge, K., & Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. In *International conference on computer vision*.

Saurer, O., Vasseur, P., Boutteau, R., Demonceaux, C., Pollefeys, M., & Fraundorfer, F. (2016). Homography based egomotion estimation with a common direction. *IEEE Tansactions on Pattern Analysis and Machine Intelligence, 39*(2), 327–341.

Sinha, S. N., Frahm, J. M., Pollefeys, M., & Genc, Y. (2006). Gpu-based video feature tracking and matching. In *Workshop on edge computing using new commodity architectures* (p 4321).

Stewénius, H., Nistér, D., Kahl, F., & Schaffalitzky, F. (2008). A minimal solution for relative pose with unknown focal length. *Image Vision Computing, 26*, 871–877.

Tian, Y., Yu, X., Fan, B., Wu, F., Heijnen, H., & Balntas, V. (2019). Sosnet: Second order similarity regularization for local descriptor learning. In *Conference on computer vision and pattern recognition*.

Turkowski, K. (1990). Transformations of surface normal vectors. In *Technical Report 22*, Apple Computer.

Wong, H. S., Chin, T. J., Yu, J., & Suter D. (2011). Dynamic and hierarchical multi-structure geometric model fitting. In *International conference on computer vision* (pp. 1044–1051). IEEE.

Yi, K. M., Verdie, Y., Fua, P. & Lepetit V.(2016). Learning to assign orientations to feature points. In *Conference on computer vision and pattern recognition*.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.