Radially-Distorted Conjugate Translations

James Pritts[1]   Zuzana Kukelova[1]   Viktor Larsson[2]   Ondřej Chum[1]

Visual Recognition Group, CTU in Prague[1]   Centre for Mathematical Sciences, Lund University[2]

Abstract

This paper introduces the first minimal solvers that jointly solve for affine-rectification and radial lens distortion from coplanar repeated patterns. Even with imagery from moderately distorted lenses, plane rectification using the pinhole camera model is inaccurate or invalid. The proposed solvers incorporate lens distortion into the camera model and extend accurate rectification to wide-angle imagery, which is now common from consumer cameras. The solvers are derived from constraints induced by the conjugate translations of an imaged scene plane, which are integrated with the division model for radial lens distortion. The hidden-variable trick with ideal saturation is used to reformulate the constraints so that the solvers generated by the Gröbner-basis method are stable, small and fast.

Rectification and lens distortion are recovered from either one conjugately translated affine-covariant feature or two independently translated similarity-covariant features. The proposed solvers are used in a RANSAC-based estimator, which gives accurate rectifications after few iterations. The proposed solvers are evaluated against the state-of-the-art and demonstrate significantly better rectifcations on noisy measurements. Qualitative results on diverse imagery demonstrate high-accuracy undistortion and rectification.

1. Introduction

Scene-plane rectification is used in many classic computer-vision tasks, including single-view 3D reconstruction, camera calibration, grouping coplanar symmetries, and image editing [26, 21, 15]. In particular, the affine rectification of a scene plane transforms the camera's principal plane so that it is parallel to the scene plane. This restores the affine invariants of the imaged scene plane, which include parallelism of lines and translational symmetries [9, 21]. There is only an affine transformation between the affine-rectified imaged scene plane and its real-world counterpart. The removal of the effects of perspective imaging is helpful to understanding the geometry of the scene plane.

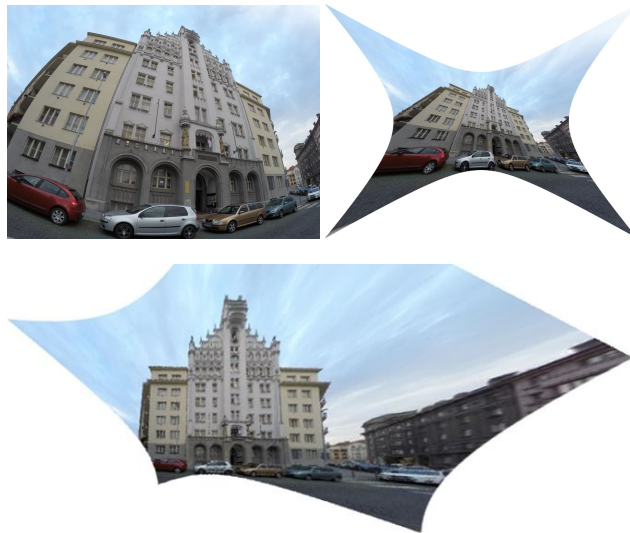Wide-angle imagery that has significant lens distortion

Figure 1: Input (top left) is a distorted view of a scene plane, and the outputs (top right, bottom) are the the undistorted and rectified scene plane. The method is fully automatic.

is common since consumer photography is now dominated by mobile-phone and GoPro-type cameras. High-accuracy rectification from wide-angle imagery is not possible with only pinhole camera models [11, 25]. Lens distortion can be estimated by performing a camera calibration apriori, but a fully automated method is desirable.

Several state-of-the-art planar-rectification methods assume a pinhole camera model, which ignores the effect of lens distortion [1, 4, 15, 27]. Pritts *et al.* [21] attempt to upgrade the pinhole camera model with radial lens distortion by giving an initial guess of the scene plane's rectification that is consistent with a pinhole camera to a non-linear optimization that incorporates a lens-distortion model. However, even with relaxed thresholds, a robust estimator (*i.e.* RANSAC) will discard measurements that capture the most extreme effects of lens distortion, especially around the boundary of the image, since these measurements are not consistent with the pinhole-camera assumption. Thus, failing to account for lens distortion while labeling the measurements as outliers, as done during a RANSAC iteration,
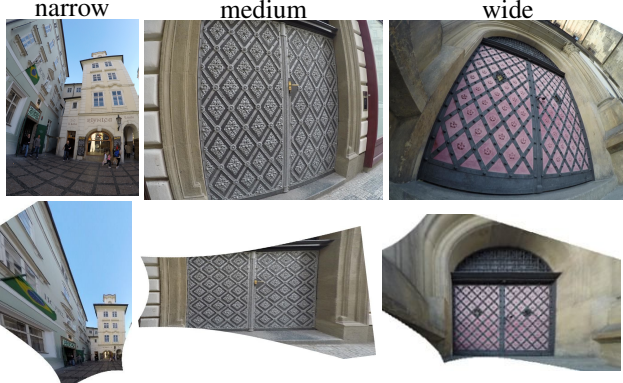
Figure 2: *GoPro Hero 4 imagery.* (top row) Input images taken at different field-of-view settings. (bottom row) Rectified results.

can give biased fits that underestimate the camera's lens-distortion [11], which, in turn, reduces rectification accuracy.

This paper introduces the first minimal solvers that jointly solve for the affine rectification of an imaged scene plane and a camera's radial lens distortion. The solvers are derived from constraints induced by the conjugate translations of an imaged scene plane (see Sec. 2 for details), which are integrated with the division model of radial distortion [7]. Despite the simple formulation of the division model, it is accurate for even wide-angle lenses [7]. In addition, the solvers estimate the vanishing translation direction of the corresponded points used for input.

Two types of solvers are introduced: *one-direction solvers*, which require 3 coplanar point correspondences that translate in the same direction, and *two-direction solvers*, which require 4 coplanar point correspondences, 2 of which translate in one-direction and the remaining 2 in a different direction. Covariant feature detectors are used to extract the needed point correspondences [14, 16, 18, 19, 24]. The solvers are used in a RANSAC-based framework for robust rectification estimation. With one or two-correspondence sampling, an accurate undistortion and rectification is quickly recovered, even for difficult scenes.

Fitzgibbon used a one-parameter division model to develop a minimal solver for jointly estimating lens distortion with a fundamental matrix or homography [7]. Kukelova *et al.* [11] proposed an extension to [7] for homographies to model two-views from cameras with different radial lens distortions. These two-view solvers can jointly estimate lens distortion and conjugate translations, but are overparameterized for this task, which can result in inaccurate estimates as is shown by the synthetic experiments in Sec. 5. Wildenauer *et al.* [25] and Antunes *et al.* [2] are two methods that use constraints induced by imaged parallel lines to

jointly solve for their vanishing point and the division model parameter, but but both require a multi-model estimation to recover scene-plane rectification (*i.e.* 2 consistent vanishing points).

The systems of polynomial equations induced from the constraints arising from joint estimation of conjugate translation with the division-model parameter are solved using an algebraic method based on Gröbner bases. Automated solver-generators using the Gröbner basis method [10, 12] were recently used to generate solvers for several problems in multi-view geometry [10, 13, 12, 11]. However, straightforward application of an automated solver-generator to the proposed problem resulted in unstable solvers (see Sec. 5). Therefore, we transformed the constraints to simplify the structure of the systems of polynomial equations, while explicitly accounting for the parasitic solutions that arose from the new formulation. The new formulation resulted in solvers with increased stability and speed.

The problem of rectification is closely coupled with the detection of coplanar repeats in a classic chicken-and-egg scenario: rectification is easy if the repeats are grouped, and repeats are more easily grouped if the affine invariants of the rectified plane are available [21]. Most methods tentatively group repeats from their local texture, which is verified later by a hypothesized rectification. Methods using this approach include Schaffalitzky *et al.* [23], which, similar to the solvers proposed in this paper, uses constraints induced by conjugate translations to recover the scene-plane's vanishing line, and Chum *et al.* [4], which uses the constraint that coplanar repeats are equiareal in the scene-plane's affine-rectified image. None of these methods account for lens distortion, and do not perform well on imagery with significant lens distortion (see Sec. 6).

2. Problem Formulation

Assume that the scene plane π and a camera's image plane π' are related point-wise by the homography P, so that $\alpha_i \mathbf{x}'_i = \mathrm{P} \mathbf{X}_i$, where α_i is a scalar, $\mathbf{X}_i \in \pi$ and $\mathbf{x}'_i \in \pi'$. Let $\mathbf{X}_i$ and $\mathbf{X}'_i$ be two points on the scene plane π such that $\mathbf{X}'_i - \mathbf{X}_i = \mathbf{t}$. By encoding $\mathbf{t}$ in the homogeneous translation matrix T, the points $\mathbf{X}_i$ and $\mathbf{X}'_i$ as imaged by camera P can be expressed as

$$\alpha_i \mathbf{x}'_i = \mathrm{P} \mathbf{X}'_i = \mathrm{PT} \mathbf{X}_i = \mathrm{PTP}^{-1} \mathbf{x}_i = \mathrm{H_u} \mathbf{x}_i, \qquad (1)$$

where the homography $\mathrm{H_u} = \mathrm{PTP}^{-1}$ is called a conjugate translation because of the form of its matrix decomposition and points $\mathbf{x}_i$ and $\mathbf{x}'_i$ are in correspondence with respect to the conjugate translation $\mathrm{H_u}$, which we denote $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ [9, 23]. Decomposing $\mathrm{H_u}$ into its projective components

gives

$$\alpha_i \mathbf{x}_i' = \mathtt{H_u} \mathbf{x}_i = \left[\mathtt{PI_3P}^{-1} + \mathtt{P} \begin{pmatrix} t_x \\ t_y \\ 1 \end{pmatrix} \left[\mathtt{P}^{-\top} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right]^{\top} \right] \mathbf{x}_i$$

$$= [\mathtt{I_3} + s_i^{\mathbf{u}} \mathbf{u} \mathbf{l}^{\top}] \cdot \mathbf{x}_i \qquad (2)$$

where $\mathtt{I_3}$ is the 3×3 identity matrix, and

- line $\mathbf{l}$ is the imaged scene plane's vanishing line,
- point $\mathbf{u}$ is the vanishing direction of translation, which must meet the vanishing line $\mathbf{l}$, *i.e.*, $\mathbf{l}^{\top} \mathbf{u} = 0$,
- and scalar $s_i^{\mathbf{u}}$ is the magnitude of translation in the direction $\mathbf{u}$ for the point correspondence $\tilde{\mathbf{x}}_i \leftrightarrow \tilde{\mathbf{x}}_i'$ [23].

Note that (2) holds only for points projected by a pinhole camera viewing a scene plane, which is parameterized by the homography $\mathtt{P}$ as defined above. For every real camera, some amount of radial distortion is always present, so for (2) to hold, the measured image points $\tilde{\mathbf{x}}_i$ and $\tilde{\mathbf{x}}_i'$ must first be undistorted. We use the one-parameter division model to parameterize the radial lens distortion [7], which has the form

$$f(\tilde{\mathbf{x}}_{\mathbf{i}}, \lambda) = \left(\tilde{x}_i, \tilde{y}_i, 1 + \lambda(\tilde{x}_i^2 + \tilde{y}_i^2) \right)^{\top}, \qquad (3)$$

where the distortion center is given; *i.e.*, $\tilde{x}_i, \tilde{y}_i$ are the center-subtracted measurements from a feature detector.

In this work we incorporate constraints induced by a conjugate translation as derived in (2) with the division model defined in (3) to accurately rectify imaged scene planes from lens-distorted cameras. Since the unknown division model parameter is exclusively encoded in the homogeneous coordinate, the relation for conjugate translations can be directly augmented to model lens distortion, namely,

$$\alpha_i f(\tilde{\mathbf{x}}_i', \lambda) = \mathtt{H_u} f(\tilde{\mathbf{x}}_i, \lambda) = [\mathtt{I_3} + s_i^{\mathbf{u}} \mathbf{u} \mathbf{l}^{\top}] \cdot f(\tilde{\mathbf{x}}_i, \lambda), \quad (4)$$

where α_i is some non-zero scalar, and $\tilde{\mathbf{x}}_i \leftrightarrow \tilde{\mathbf{x}}_i'$ is a point correspondence.

3. Solvers

The model for radially-distorted conjugate translations in (4) defines the unknown geometric quantities: (i) division-model parameter λ, (ii) scene-plane vanishing line $\mathbf{l} = \left(l_1, l_2, l_3 \right)^{\top}$, (iii) vanishing translation direction $\mathbf{u} = \left(u_1, u_2, u_3 \right)^{\top}$ (see Sec. 3.2 for the two-direction extensions), (iv) scale of translation $s_i^{\mathbf{u}}$ for correspondence $\tilde{\mathbf{x}}_i \leftrightarrow \tilde{\mathbf{x}}_i'$, (v) and the scalar parameter α_i.

The solution for the vanishing line $\mathbf{l}$ is constrained to the affine subspace $l_3 = 1$ of the real-projective plane, which makes it unique. This inhomogeneous choice of $\mathbf{l}$ is unable to represent the pencil of lines that pass through the origin. If this degeneracy is encountered, then the scale of $\mathbf{l}$

is fixed by setting $l_2 = 1$ instead. Solver variants for both constraints are generated for all of the proposed solvers. In practice, this degeneracy is rarely encountered. If the $l_3 = 1$ solver variant suffers from bad numerical conditioning, then the $l_2 = 1$ variant can be activated and its solutions tested for consensus with the measurements (see Sec. 4). Without loss of generality the derivations below assume that $l_3 = 1$.

The vanishing direction $\mathbf{u}$ must meet the vanishing line $\mathbf{l}$, which defines a subspace of solutions for $\mathbf{u}$. The magnitude of $\mathbf{u}$ is set to the translation scale $s_1^{\mathbf{u}}$ of the first correspondence, which defines a unique solution

$$\mathbf{l}^{\top} \mathbf{u} = l_1 u_1 + l_2 u_2 + u_3 = 0 \quad \wedge \quad \|\mathbf{u}\| = s_1^{\mathbf{u}}. \quad (5)$$

The relative scale of translation $\bar{s}_i^{\mathbf{u}}$ for each correspondence $\tilde{\mathbf{x}}_i \leftrightarrow \tilde{\mathbf{x}}_i'$ with respect to the magnitude of $\|\mathbf{u}\|$ is defined so that $\bar{s}_i^{\mathbf{u}} = s_i^{\mathbf{u}}/\|\mathbf{u}\|$. Note that $\bar{s}_1^{\mathbf{u}} = 1$.

In this paper we propose four different minimal solvers for different variants of the problem of radially-distorted conjugate translations based on different translation directions and relative scales $\bar{s}_i^{\mathbf{u}}$. These variants are motivated by the types of covariant feature detectors used to extract point correspondences [14, 16, 18, 19, 24]. Each affine-covariant feature defines an affine frame, *i.e.* an ordered triplet of points. Thus, 1 affine-frame correspondence provides the 3 point correspondences that translate in the same direction with the same scale. This is sufficient input for the one-directional solvers. A visualization of the features is provided in Fig. C.4 of the supplemental material. In the case of similarity-covariant features, such as DoG [14], only a similarity frame can be constructed. A correspondence of similarity frames gives 2 point correspondences that translate jointly. Two correspondences of similarity-covariant features of different direction of the translation provide sufficient constraints for the two directional solvers.

Two *one-direction solvers* are proposed, which require 3 (2.5) coplanar point correspondences that translate in the same direction. The "3-point" solver $\mathtt{H3lu}s_\mathbf{u}\lambda$ assumes that two of the point correspondences have the same scale of translation (i.e. $\bar{s}_1^{\mathbf{u}} = \bar{s}_2^{\mathbf{u}} = 1$), and the third point correspondence has an unknown relative scale of the translation $\bar{s}_3^{\mathbf{u}}$. The "2.5-point" solver $\mathtt{H2.5lu}\lambda$ assumes that all 3 point correspondences have the same relative scales of translation, i.e. $\bar{s}_1^{\mathbf{u}} = \bar{s}_2^{\mathbf{u}} = \bar{s}_3^{\mathbf{u}} = 1$.

In addition, two *two-direction solvers* are proposed, which require 4 (3.5) coplanar point correspondences, 2 of which translate in one-direction $\mathbf{u}$ and the remaining 2 in a different direction $\mathbf{v}$. Here the "4-point" solver $\mathtt{H4luv}s_\mathbf{v}\lambda$ assumes that the first two point correspondences translate in the direction $\mathbf{u}$ with the same relative scale of translation, i.e., $\bar{s}_1^{\mathbf{u}} = \bar{s}_2^{\mathbf{u}} = 1$. The remaining two point correspondence translate in the direction $\mathbf{v}$ with arbitrary translation magnitudes, i.e., the relative scales of translations of these two correspondences with respect to $\|\mathbf{v}\| = s_3^{\mathbf{v}}$ are $\bar{s}_3^{\mathbf{v}} = 1$

and an unknown relative scale $\bar{s}_4^{\mathbf{u}}$.

The "3.5-point" H3.5**luv**λ solver assumes that the relative scales $\bar{s}_1^{\mathbf{u}} = \bar{s}_2^{\mathbf{u}} = 1$ with respect to $\|\mathbf{u}\| = s_1^{\mathbf{u}}$ and $\bar{s}_3^{\mathbf{v}} = \bar{s}_4^{\mathbf{v}} = 1$ with respect to $\|\mathbf{v}\| = s_3^{\mathbf{v}}$.

In all proposed solvers the scalar values α_i are eliminated from (4). This is done by multiplying (4) by the skew-symmetric matrix $[f(\tilde{\mathbf{x}}_i', \lambda)]_\times$. The fact that the join of a point $\mathbf{x}_i$ with itself $[\mathbf{x}_i]_\times \mathbf{x}_i$ is $\mathbf{0}$ gives,

$$
\begin{bmatrix} 0 & -\tilde{w}_i' & \tilde{y}_i' \\ \tilde{w}_i' & 0 & -\tilde{x}_i' \\ -\tilde{y}_i' & \tilde{x}_i' & 0 \end{bmatrix}
$$
$$
\times \begin{bmatrix} 1 + \bar{s}_i^{\mathbf{u}} u_1 l_1 & \bar{s}_i^{\mathbf{u}} u_1 l_2 & \bar{s}_i^{\mathbf{u}} u_1 \\ \bar{s}_i^{\mathbf{u}} u_2 l_1 & 1 + \bar{s}_i^{\mathbf{u}} u_2 l_2 & \bar{s}_i^{\mathbf{u}} u_2 \\ \bar{s}_i^{\mathbf{u}} u_3 l_1 & \bar{s}_i^{\mathbf{u}} u_3 l_2 & 1 + \bar{s}_i^{\mathbf{u}} u_3 \end{bmatrix} \begin{pmatrix} \tilde{x}_i \\ \tilde{y}_i \\ \tilde{w}_i \end{pmatrix} = \mathbf{0},
$$
(6)

where $\tilde{w}_i = 1 + \lambda(\tilde{x}_i^2 + \tilde{y}_i^2)$ and $\tilde{w}_i' = 1 + \lambda(\tilde{x}_i'^2 + \tilde{y}_i'^2)$. The matrix equation in (6) contains three polynomial equations from which only two are linearly independent, since the skew-symmetric matrix $[f(\tilde{\mathbf{x}}_i', \lambda)]_\times$ is rank two.

To solve the systems of polynomial equations resulting from the presented problems, we use the Gröbner basis method [6]. To generate efficient solvers we used the automatic generator of Gröbner basis solvers proposed in [10, 12]. However, for our problems the coefficients of the input equations are not fully independent. This means that using the default settings for the automatic generator [10, 12] that initialize the coefficients of equations by random values from $\mathbb{Z}_p$ does not lead to correct solvers. To obtain working Gröbner basis solvers, one has to create correct problems instances with values from $\mathbb{Z}_p$ for the automatic generator initialization.

The straightforward application of the automatic generator [10, 12] to the needed constraints with correct coefficients from $\mathbb{Z}_p$ resulted in large templates and unstable solvers, especially for the two-direction problems. The Gröbner basis solvers generated for the original constraints have template matrices with sizes 80×84, 74×76, 348×354, and 730×734 for the H2.5**lu**λ, H3**lu**$s_{\mathbf{u}}\lambda$, H3.5**luv**λ and H4**luv**$s_{\mathbf{v}}\lambda$ problems, respectively. Therefore, we use the hidden-variable trick [6] to eliminate the vanishing translation directions together with ideal saturation [13] to eliminate parasitic solutions. The reformulated constraints are simpler systems in only 3 or 4 unknowns, and the solvers generated by the Gröbner basis method are smaller and more stable. The reduced eliminiation template sizes are also summarized in Sec. B of the supplemental material. Next, we describe the solvers based on the hidden-variable trick in more detail.

3.1. One-Direction Solvers

For the "3-point" one-direction H3**lu**$s_{\mathbf{u}}\lambda$ solver we have $\bar{s}_1^{\mathbf{u}} = \bar{s}_2^{\mathbf{u}} = 1$. Therefore the constraints (6) result in two

	H2**lu**	H2.5**lu**λ	H3**lu**$s_{\mathbf{u}}\lambda$	H3.5**luv**λ	H4**luv**$s_{\mathbf{v}}\lambda$	H4**l**γ	H5λ	H5$\lambda_1\lambda_2$
Reference	[23]					[4]	[7]	[10]
Distortion		✓	✓	✓	✓		✓	✓
H$_\infty$	✓	✓	✓	✓	✓	✓		
# points	2	2.5	3	3.5	4	4	5	5
# solutions	1	4	2	6	4	1	18	5

Table 1: Proposed solvers (grey) vs. state-of-the-art.

pairs of linearly independent equations without the scale parameter $\bar{s}_i^{\mathbf{u}}$ for $i = 1, 2$, and two linearly independent equations with an unknown relative scale $\bar{s}_3^{\mathbf{u}}$ for the third point correspondence, i.e., $i = 3$. Additionally, we have the orthogonality constraint in (5). All together we have seven equations in seven unknowns $(l_1, l_2, u_1, u_2, u_3, \bar{s}_3^{\mathbf{u}}, \lambda)$.

Note, that these equations are linear with respect to the vanishing translation direction $\mathbf{u}$. Therefore, we can rewrite the seven equations as

$$
\mathbf{M}(l_1, l_2, \bar{s}_3^{\mathbf{u}}, \lambda) \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ 1 \end{pmatrix} = \mathbf{0}
$$
(7)

where $\mathbf{M}(l_1, l_2, \bar{s}_3^{\mathbf{u}}, \lambda)$ is a 7×4 matrix which elements are polynomials in $(l_1, l_2, \bar{s}_3^{\mathbf{u}}, \lambda)$.

Since $\mathbf{M}(l_1, l_2, \bar{s}_3^{\mathbf{u}}, \lambda)$ has a null vector, it must be rank deficient. Therefore, all the 4×4 sub-determinants of $\mathbf{M}(l_1, l_2, \bar{s}_3^{\mathbf{u}}, \lambda)$ must equal zero. This results in $\binom{7}{4} = 35$ polynomial equations which only involve four unknowns.

Unfortunately, the formulation (7) introduces a one-dimensional family of false solutions. These are not present in the original system and corresponds to solutions where the first three columns of $\mathbf{M}$ become rank deficient. In this case there exist null vectors to $\mathbf{M}$ where the last element of the vector is zero, i.e. not on the same form as in (7).

These false solutions can be removed by saturating any of the 3×3 sub-determinants from the first three columns of $\mathbf{M}$. The matrix $\mathbf{M}$ has the following form,

$$
\mathbf{M}(l_1, l_2, \bar{s}_3^{\mathbf{u}}, \lambda) = \begin{bmatrix} m_{11} & m_{12} & 0 & m_{14} \\ m_{21} & m_{22} & 0 & m_{24} \\ m_{31} & 0 & m_{33} & m_{34} \\ m_{41} & 0 & m_{43} & m_{44} \\ m_{51} & m_{52} & 0 & m_{54} \\ m_{61} & 0 & m_{63} & m_{64} \\ l_1 & l_2 & 1 & 0 \end{bmatrix}
$$
(8)

where m_{ij} are polynomials in $l_1, l_2, \bar{s}_3^{\mathbf{u}}$ and λ. We choose to saturate the 3×3 sub-determinant corresponding to the first, second and last row since it reduces to only the top-left 2×2

sub-determinant, i.e. $m_{11}m_{22} - m_{12}m_{21}$, which is only a quadratic polynomial in the unknowns. The other 3×3 determinants are more complicated and leads to larger polynomial solvers. Using the saturation technique from Larsson *et al.* [13] we were able to create a polynomial solver for this saturated ideal. The size of the elimination template is 24×26. Note that without using the hidden-variable trick the elimination template was 74×76.

For the H2.5$\mathbf{lu}\lambda$ solver we can use the same hidden-variable trick. In this case $\bar{s}_1^{\mathbf{u}} = \bar{s}_2^{\mathbf{u}} = \bar{s}_3^{\mathbf{u}} = 1$ and therefore the matrix M in (7) contains only three unknowns l_1, l_2 and λ. The minimal number of point correspondences necessary to solve this problem is 2.5. Therefore, for this problem we can drop one of the equations from (6), e.g., for $i = 3$, and the matrix M in (7) has size 6×4. In this case all 4×4 sub-determinants of M result in 15 equations in 3 unknowns.

Similar to the 3 point case, this introduces a one-dimensional family of false solutions. The matrix M has a similar structure as in (8) and again it is sufficient to saturate top-left 2×2 sub-determinant. For this formulation we were able to create a solver with template size 14×18 (compared with 80×84 without using hidden-variable trick)

3.2. Two-Direction Solvers

In the case of the two-direction H4$\mathbf{luv}s_{\mathbf{v}}\lambda$ solvers, the input equations for two vanishing translation directions $\mathbf{u} = (u_1, u_2, u_3)^\top$ and $\mathbf{v} = (v_1, v_2, v_3)^\top$ can be separated into two sets of equations, i.e., the equations containing $\mathbf{u}$ and the equations containing $\mathbf{v}$. Note that in this case we have two equations of the form (5), i.e., the equation for the direction $\mathbf{u}$ and the equation for the direction $\mathbf{v}$ and we have an unknown relative scale $\bar{s}_4^{\mathbf{v}}$. Therefore, the final system of 10 equations in 10 unknowns can be rewritten using two matrix equations as

$$\mathsf{M}_1(l_1, l_2, \lambda) \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ 1 \end{pmatrix} = \mathbf{0}, \quad \mathsf{M}_2(l_1, l_2, \bar{s}_4^{\mathbf{v}}, \lambda) \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ 1 \end{pmatrix} = \mathbf{0} \quad (9)$$

where M_1 and M_2 are 5×4 matrices where the elements are polynomials in (l_1, l_2, λ) and $(l_1, l_2, \bar{s}_4^{\mathbf{v}}, \lambda)$ respectively.

Again all 4×4 sub-determinants of M_1 and M_2 must concurrently equal zero. This results in $5 + 5 = 10$ polynomial equations in four unknowns $(l_1, l_2, \bar{s}_4^{\mathbf{v}}, \lambda)$. In this case, only 39 additional false solutions arise from the hidden-variable trick. The matrices M_1 and M_2 have a similar structure as in (8) and again it is sufficient to saturate the top-left 2×2 sub-determinants to remove the extra solutions. By saturating these determinants we were able to create a solver with template size 76×80 (previously 730×734).

Finally, for the "3.5-point" two-direction H3.5$\mathbf{luv}\lambda$ solver $\bar{s}_1^{\mathbf{u}} = \bar{s}_2^{\mathbf{u}} = 1$ and $\bar{s}_3^{\mathbf{v}} = \bar{s}_4^{\mathbf{v}} = 1$ so we can drop one of the equations from the constraint (6), e.g., for $i = 4$.

Therefore, the matrix M_2 from (9) has size 4×4 and it contains only 3 unknowns (l_1, l_2, λ). In this case all 4×4 sub-determinants of M_1 and M_2 result in $5 + 1 = 6$ polynomial equations in three unknowns (l_1, l_2, λ).

For this case we get 18 additional false solutions. Investigations in Macaulay2 [8] revealed that for this particular formulation it was sufficient to only saturate the top-left 2×2 sub-determinant of M_1 and the top-left element of M_2. Saturating these we were able to create a polynomial solver with a template size of 54×60 (previously 348×354).

4. Robust Estimation

The solvers are used in a LO-RANSAC-based robust-estimation framework [5]. Affine-covariant features are extracted from the image for input to the solvers. Affine-covariant features are highly repeatable on the same imaged scene texture with respect to significant changes of viewpoint and illumination [17]. Their proven robustness in the multi-view correspondence problem makes them good candidates for representing the local geometry of repeated textures. In particular, for the real-image experiments in Sec. 6, we use the Maximally-Stable Extremal Region and Hesssian-Affine detectors [16, 18]. The detections are parameterized as 3 distinct points, which define an affine coordinate frame in the image space [20]. These detections are visualized in Fig. C.4 of the supplemental material.

Affine frames are labeled as repeated texture based on the similarity of their appearance, which is given by the RootSIFT embedding of the image patch local to the affine frame [3, 14]. The RootSIFT descriptors are agglomeratively clustered, which establishes pair-wise tentative correspondences among the connected components linked by the clustering. Each appearance cluster has some proportion of its members that correspond to affine frames that give the geometry of imaged repeated scene content, which are the *inliers* of that appearance cluster. The remaining affine frames are the *outliers*.

LO-RANSAC samples pairs of affine frames from the appearance cluster, which are inputted to the proposed minimal solvers. Each pair of affine frames across all appearance clusters has an equi-probable chance of being drawn. The consensus with the minimal sample is measured by the number of pairs of affine frames within appearance groups that are consistent with the hypothesized model, normalized by the size of each respective group. A non-linear optimizer following [21] is used as the local optimization step of the LO-RANSAC estimator.

5. Synthetic Experiments

The proposed solvers are evaluated across several benchmarks on synthetic data against state-of-the-art solvers. Included in the benchmarks are two single-view solvers: H2$\mathbf{lu}$
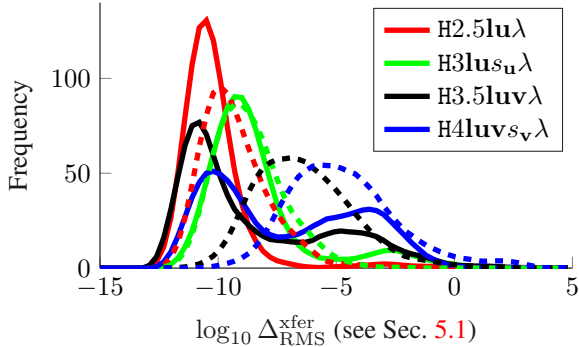
Figure 3: *Stability study.* Hidden-variable trick solvers are solid; standard solvers are dashed. The $\log_{10}$ transfer error is reported. The hidden-variable trick increases stability.

[23], which also incorporates constraints from conjugate translations, and H4lγ [4], which solves for rectification and change-of-scale, and also two full-homography and radial distortion solvers, H5λ [7] and H5$\lambda_1\lambda_2$ [11], which we use for conjugate translation and lens-istortion estimation. The bench of state-of-the-art solvers is summarized in Table 1).

The benchmarks are evaluated for 1000 synthetic images of 3D scenes with known ground-truth parameters. A camera with a random but realistic focal length is randomly oriented and positioned with respect to a 10x10 square meter scene plane such that the plane is mostly in the camera's field-of-view. Image resolution is set to 1000x1000 pixels. Conjugately translated affine frames are generated on the scene plane such that their scale with respect to the scene plane is realistic. This modeling choice reflects the use of affine-covariant feature detectors for real images. The conjugately translated features are distorted according to the division model, and, for the sensitivity experiments, isotropic white noise is added to the distorted affine frames at increasing levels. Performance is characterized by the relative error of the estimated distortion parameter and by the transfer and warp errors, which measure the accuracies of the estimated conjugate translation and rectification (see Sec. 5.1 - 5.4). The proposed solvers have an average solve time from 0.3 to 2 milliseconds over the 1000 synthetic scenes (see also Sec. B of the supplemental material).

5.1. Transfer Error

The geometric transfer error jointly measures the accuracy of an the estimated conjugate translation and lens distortion. The scene plane is tesselated by a 10x10 square grid of points $\{\mathbf{X}_i\}$. Let the translation on the scene plane induced by the noiseless pre-images of the point correspondences used to estimate $\hat{\mathbf{H}}_\mathbf{u}$ and $\hat{\lambda}$ be $\mathbf{t}$. Then the grid points are translated by $\mathbf{t}/\|\mathbf{t}\|$ to $\{\mathbf{X}'_i\}$. The grid and its translation are imaged by the ground-truth lens-distorted cam-

era parameterized by matrix P and division-model parameter λ. The imaged grid is given by $\tilde{\mathbf{x}}_i = f^d(\mathtt{P}\mathbf{X}_i, \lambda)$ and the translated grid by $\tilde{\mathbf{x}}'_i = f^d(\mathtt{P}\mathbf{X}'_i, \lambda)$, where f^d is the the function that transforms from pinhole points to radially-distorted points. Then the geometric transfer error is defined as

$$\Delta_i^{\mathrm{xfer}} = d(f^d([\mathtt{I}_3 + \frac{1}{\|\mathbf{t}\|}(\hat{\mathbf{H}}_\mathbf{u} - \mathtt{I}_3)]f(\tilde{\mathbf{x}}_i, \hat{\lambda}_1), \hat{\lambda}_2), \tilde{\mathbf{x}}'_i),\tag{10}$$

where $d(\cdot, \cdot)$ is the Euclidean distance. All solvers except H5$\lambda_1\lambda_2$ have the constraint that $\hat{\lambda}_1 = \hat{\lambda}_2$ [11]. The root-mean-square of transfer errors $\Delta_{\mathrm{RMS}}^{\mathrm{xfer}}$ for correspondences $\{(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}'_i)\}$ is reported. For two-direction solvers, the transfer error in the second direction is included in $\Delta_{\mathrm{RMS}}^{\mathrm{xfer}}$. The transfer error is used in the stability study, where the solvers are tested over varying division model parameters and in the sensitivity study, where the solvers are tested over varying noise levels with fixed division model parameter. The solver H4lγ of [4] does not estimate conjugate translations, so it is not reported. For a derivation of (10) see Sec. A in the supplementary material.

5.2. Numerical Stability

The stability study measures the RMS transfer error of solvers (see Sec. 5.1) for noiseless affine-frame correspondences across realistic scene and camera configurations generated as described in the introduction to this section. The ground-truth parameter of the division model λ is drawn uniformly at random from the interval $[-6, 0]$. For a reference, the division parameter of $\lambda = -4$ is typical for wide field-of-view cameras like the GoPro where the image is normalized by $\frac{1}{width+height}$. Fig. 3 reports the histogram of $\log_{10}$ RMS transfer errors. For all new solvers we evaluate a solver generated from constraints derived with (solid histogram) and without (dashed histogram) the hidden-variable trick. The hidden-variable trick significantly improves the stability of the proposed solvers. The increased stabilities of the hidden-variable solvers most likely result from the reduced size of the G-J elimination problems needed by these solvers. The hidden-variable solvers are used for the remainder of the experiments.

5.3. Noise Sensitivity

The proposed and state-of-the-art solvers solvers are tested with increasing levels of white noise added to the affine correspondences induced by the ground-truth conjugate translation and lens distortion parameter. The white noise is parameterized by the standard-deviation of a zero-mean isotropic Gaussian distribution, and the solvers are tested at noise levels of $\sigma \in \{0.1, 0.5, 1, 2\}$. The ground truth division model parameter is set to $\lambda = -4$, which is typical for GoPro-type imagery. The solvers are wrapped in
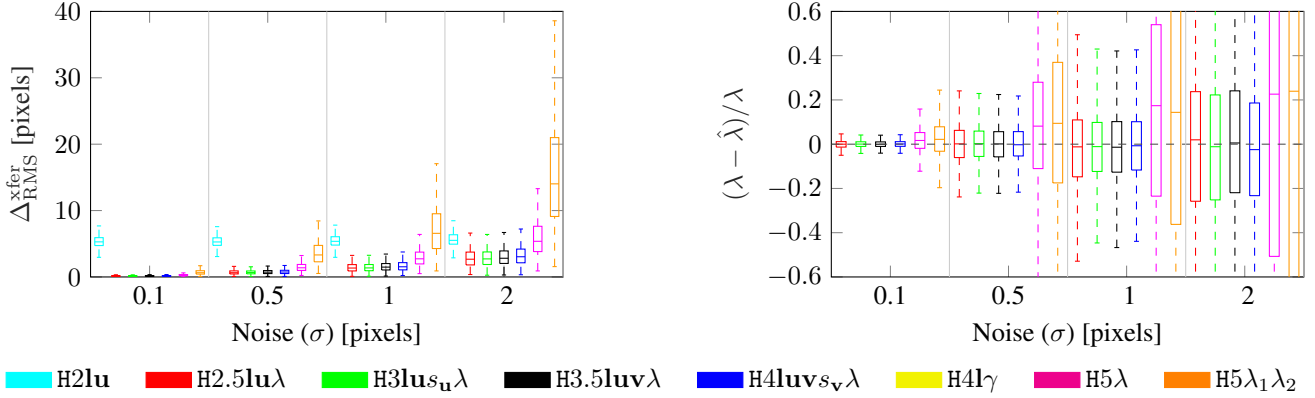
Figure 4: Comparison of the transfer error (left, see Sec. 5.1) and the relative radial distortion error (right) after 25 iterations of a simple RANSAC for different solvers over increasingly noisy measurements for 1000 scenes.
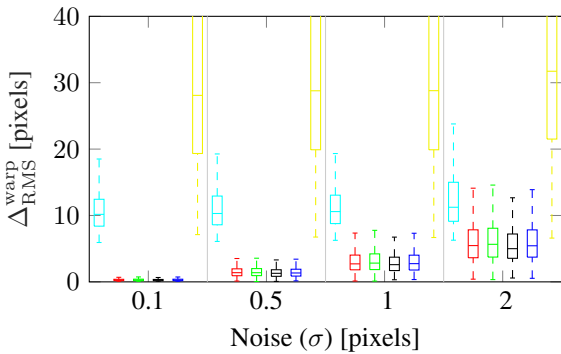

Figure 5: Warp-error comparison (see Sec. 5.4) after 25 iterations of a simple RANSAC for different solvers over increasingly noisy measurements for 1000 scenes.

a simple RANSAC-loop, which minimizes the RMS transfer error $\Delta_{\mathrm{RMS}}^{\mathrm{xfer}}$ over 25 sampled affine-frame correspondences. Results are calculated from the estimate given by RANSAC and summarized for the 1000 generated scenes as boxplots. The interquartile range is contained within the extents of a box, and the median is the horizontal line dividing the box.

As shown in Fig. 4 the proposed solvers give the most accurate joint estimation of conjugate translation and division model parameter as measured by the RMS transfer error $\Delta_{\mathrm{RMS}}^{\mathrm{xfer}}$ and relative error of estimated divsion model parameter. As expected, the minimal parameterization of the radially-distorted conjugate translation solvers—H2.5luλ, H3lu$s_{\mathbf{u}}\lambda$, H3.5luvλ, H4luv$s_{\mathbf{v}}\lambda$—show significantly less sensitivity to noise than the overparameterized radially-distorted homography solvers H5λ and H5$\lambda_1\lambda_2$ for both measures. The solver H2lu shows significant bias (see the transfer error boxplots in Fig. 4) since it does not model lens distortion.

5.4. Warp Error

Since the accuracy of scene-plane rectification is a primary concern, we also report the warp error for rectifying homographies proposed by Pritts *et al.* [22], which we augment with the division model for radial lens distortion. A rectifying homography $\mathtt{H}_\infty$ of an imaged scene plane is constructed from its vanishing line $\mathbf{l}$ (see [9]). A round trip between the image space and rectified space is made by undistorting and rectifying imaged coplanar points by the estimated lens distortion $\hat{\lambda}$ and rectifying homography $\hat{\mathtt{H}}_\infty$ and then re-warping and distorting the rectified points into the image by a synthetic camera constructed from the ground-truth lens distortion λ and rectifying homography $\mathtt{H}_\infty$. Ideally, the synthetic camera constructed from the truth would project the undistorted and rectified points onto the original points.

Note that there is an affine ambiguity, denoted A, between $\hat{\mathtt{H}}_\infty$ and $\mathtt{H}_\infty$, which is folded into the expression for the synthetic camera, namely $\mathtt{P}(\mathtt{A}) = (\mathtt{A}\mathtt{H}_\infty)^{-1}$, and estimated during computation of the warp error,

$$\Delta^{\mathrm{warp}} = \min_{\hat{\mathtt{A}}} \sum_i d^2(\tilde{\mathbf{x}}_i, f^d(\mathtt{P}(\hat{\mathtt{A}})\hat{\mathtt{H}}_\infty f(\tilde{\mathbf{x}}_i, \hat{\lambda})), \hat{\lambda}), \quad (11)$$

where $d(\cdot, \cdot)$ is the Euclidean distance, and $\{\tilde{\mathbf{x}}_i\}$ are the imaged grid points of the scene-plane tesselation as defined in Sec. 5.1. The root mean square warp error for $\{\tilde{\mathbf{x}}_i\}$ is reported and denoted as $\Delta_{\mathrm{RMS}}^{\mathrm{warp}}$. The vanishing line is not directly estimated by solvers H5λ of [7] and H5$\lambda_1\lambda_2$ of [11], so they are not reported.

The proposed solvers—H2.5luλ, H3lu$s_{\mathbf{u}}\lambda$, H3.5luvλ, H4luv$s_{\mathbf{v}}\lambda$—estimate rectifications with less than 5 pixel RMS warp error $\Delta_{\mathrm{RMS}}^{\mathrm{warp}}$, even at the 2 pixel noise level, see Fig. 5. The need to model radial lens distortion is shown by the biased fits for the solvers H2lu, H4γ.
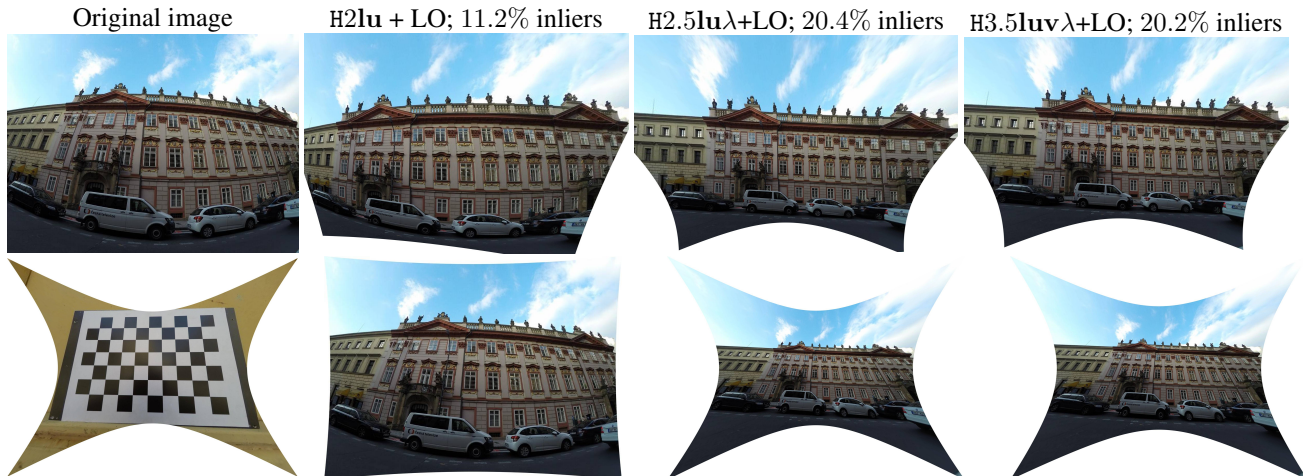
Figure 6: *GoPro Hero 4 at the wide setting for different solvers.* Results from LO-RANSAC (see Sec. 4) for H2**lu**, which omits distortion, and the proposed solvers H2.5**lu**λ and H3.5**luv**λ. The top row has rectifications after local optimization (LO); The bottom row has undistortions estimated from the best *minimal* sample. LO-RANSAC cannot recover from the poor initializations by H2**lu** (column 2). The proposed solvers in columns 3 and 4 give a correct rectification. The bottom left has a chessboard undistorted using the division parameter estimated from the building facade by H2.5**lu**λ+LO.

6. Real Images

In the qualitative experiments on real images shown in Figs. 1 and 2, we tested the proposed solvers on Go-Pro4 Hero 4 images with increasing field-of-view settings, namely narrow, medium and wide, where a wider field-of-view setting generates more extreme radial distortion since the boundary of the lens is used. The proposed method generates high-quality rectifications at all the field-of-view settings. More real-image experiments, including results for cameras with radial distortions that are typical for mobile phone cameras and fisheye lenses (*e.g.*, 8mm lens) can be found in Sec. C in the supplementary material.

The experiment shown in Fig. 6 compares the performance of two of the proposed solvers, H2.5**lu**λ and H3.5**luv**λ, to H2**lu** in a state-of-the-art local-optimization (LO) framework (see Sec. 4) on an GoPro Hero 4 image at the wide field-of-view setting. The two proposed solvers accurately estimate the division-model parameter (see the undistorted reference chessboard in Fig. 6) and the rectification, while the LO-variant using the H2**lu** solver is unable to recover the lens distortion parameter. See Fig. C.2 in the supplemental material for results on an image at the medium field-of-view setting.

7. Conclusions

This paper proposes the first minimal solvers that jointly solve for the affine rectification of an imaged scene plane and a camera's radial lens distortion from coplanar repeated patterns. Rectification and radial lens distortion are recov-

ered from only one conjugately translated affine-covariant feature or two independently translated similarity-covariant features. Synthetic experiments demonstrate the good stability and superior robustness to noise with respect to measures of rectification accuracy and lens-distortion estimation of the proposed solvers as compared to the state-of-the-art. However, the polynomial constraint equations that arise from conjugate translations distorted by the division model need to be transformed with the hidden-variable trick to generate stable solvers, though. Qualitative real-image experiments demonstrate high-quality rectifications for highly-distorted wide-angle lenses, which was not possible using the state-of-the-art. Future work could include conditionally sampling the measurements during robust estimation to take into account their size, relative distance from each other, or distance from the distortion center. We expect these factors have a big impact on rectification quality, but this study was beyond scope for this paper.

8. Acknowledgements

References

[1] D. Aiger, D. Cohen-Or, and N. Mitra. Repetition maximization based texture rectification. *Computer Graphics Forum*, 31(2):439–448, 2012. 1

[2] M. Antunes, J. P. Barreto, D. Aouada, and B. Ottersten. Unsupervised vanishing point detection and camera calibration from a single manhattan image with radial distortion. In *CVPR*, July 2017. 2

[3] R. Arandjelović and A. Zisserman. Three things everyone should know to improve object retrieval. In *CVPR*, 2012. 5

[4] O. Chum and J. Matas. Planar affine rectification from change of scale. In *ACCV*, 2010. 1, 2, 4, 6

[5] O. Chum, J. Matas, and v. Obdržálek. Enhancing RANSAC by generalized model optimization. In *ACCV*, 2004. 5

[6] D. Cox, J. Little, and O. D. Using algebraic geometry. Springer, 2004. 4

[7] A. W. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *CVPR*, 2001. 2, 3, 4, 6, 7, 12

[8] D. R. Grayson and M. E. Stillman. Macaulay 2, a software system for research in algebraic geometry, 2002. 5

[9] R. I. Hartley and A. W. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004. 1, 2, 7

[10] Z. Kukelova, M. Bujnak, and T. Pajdla. Automatic generator of minimal problem solvers. In *ECCV*, 2008. 2, 4

[11] Z. Kukelova, J. Heller, B. M., and T. Pajdla. Radial distortion homography. In *CVPR*, 2015. 1, 2, 6, 7

[12] V. Larsson, K. Åström, and M. Oskarsson. Efficient solvers for minimal problems by syzygy-based reduction. In *CVPR*, 2017. 2, 4

[13] V. Larsson, K. Åström, and M. Oskarsson. Polynomial solvers for saturated ideals. In *ICCV*, 2017. 2, 4, 5

[14] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 2, 3, 5

[15] M. Lukáč, D. Sýkora, K. Sunkavalli, E. Shechtman, O. Jamriška, N. Carr, and T. Pajdla. Nautilus: Recovering regional symmetry transformations for image editing. *ACM Trans. Graph.*, 36(4):108:1–108:11, July 2017. 1

[16] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC*, 2002. 2, 3, 5

[17] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *PAMI*, 2004. 5

[18] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. volume 60, 2004. 2, 3, 5

[19] D. Mishkin, F. Radenovic, and J. Matas. Learning discriminative affine regions via discriminability. *CoRR*, abs/1711.06704, 2017. 2, 3

[20] Š. Obdržálek and J. Matas. Object recognition using local affine frames on distinguished regions. In *BMVC*, 2002. 5

[21] J. Pritts, O. Chum, and J. Matas. Detection, rectification and segmentation of coplanar repeated patterns. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 1, 2, 5

[22] J. Pritts, D. Rozumnyi, M. P. Kumar, and O. Chum. Coplanar repeats by energy minimization. In *BMVC*, 2016. 7

[23] F. Schaffalitzky and A. Zisserman. Geometric grouping of repeated elements within images. In *BMVC*, 1998. 2, 3, 4, 6

[24] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008. 2, 3

[25] H. Wildenauer and B. Micusík. Closed form solution for radial distortion estimation from a single vanishing point. In *BMVC*, 2013. 1, 2

[26] C. Wu, J. M. Frahm, and M. Pollefeys. Repetition-based dense single-view reconstruction. In *CVPR*, 2011. 1

[27] Z. Zhang, A. Ganesh, X. Liang, and Y. Ma. TILT: transform invariant low-rank textures. *IJCV*, 99(1):1–24, 2012. 1