— Supplementary Material —
A Large-Scale Homography Benchmark

Daniel Barath[1], Dmytro Mishkin[2,5], Michal Polic[2,3], Wolfgang Förstner[4], Jiri Matas[2]
[1]Computer Vision and Geometry Group, ETH Zurich, Switzerland,
[2]VRG, Faculty of Electrical Engineering, CTU in Prague, Czech Republic,
[3]CIIRC, CTU in Prague, Czech Republic, [4]University Bonn, Germany, [5]HOVER Inc.

Contents

1. Fine-tuning Correspondence Filtering

To demonstrate that the proposed dataset can be used for improving correspondence-filtering networks, we fine-tuned OANet [28]. We use only the correspondence classification loss on the training split of the dataset. We run 2 epochs with Adam optimizer and 1e-4 learning rate. The OANet+RANSAC results on the test set improved from 0.368 to 0.379 mAA. We did not explore fine-tuning other methods since most state-of-the-art networks have epipolar-geometry-specific losses and parts inside. Repurposing them to the homography-only case is outside the scope of this paper.

2. Methods in the Main Experiments

In this section, we describe the components of each algorithm compared in the main paper.

2.1. Traditional Algorithms

In all tested methods, the normalized direct linear transformation [15] (DLT) algorithm runs both on minimal and non-minimal samples to estimate homographies. The compared methods and implementations are the following.

RANSAC (OpenCV). The OpenCV implementation contains the following components in addition to the original RANSAC [11] algorithm.

1. Sample cheirality check to reject minimal samples early if the implied plane flips between the two views.
2. Levenberg-Marquardt numerical optimization minimizes the re-projection error on the final set of inliers.
3. Single-sided re-projection error, measured in the second image, is used as point-to-model residual.

LMEDS (OpenCV). The OpenCV implementation of the Least Median of Squares algorithm [23] runs the same additional components as the OpenCV RANSAC.

LSQ (OpenCV). The least-squares fitting by the normalized four-point algorithm implemented in OpenCV.

RANSAC (skimage). The RANSAC as implemented in the scikit-image library [25]. It contains the following components in addition to the original RANSAC [11] algorithm. Single-sided re-projection error, measured in the second image, is used as point-to-model residual. Unlike OpenCV RANSAC, which is implemented in optimized C++ code,

scikit-image is implemented in pure Python with help of the numpy package [14].

LO-RANSAC (kornia). LO-RANSAC [8] as implemented in the kornia library [22]. It implements the LO-RANSAC as proposed in [8] (version 2 in Section 3), where the far-the-best model is obtained by running local optimization using all inliers. Additional components:

1. Symmetric transfer error is used as point-to-model residual.
2. Unlike LO-RANSAC, the kornia library uses iterated re-weighted least squares for the local optimization.
3. Unlike the rest of the RANSAC implementations, kornia generates and evaluates hypotheses in "batches" of 1024 to make use of CPU and GPU parallelism.

LO-RANSAC+ (pydegensac). The algorithm from [18] as implemented in pydegensac package.It uses truncated quadratic cost function and fast local optimization scheme using a subset of inlier sets.

LO-RANSAC+ with LAF (pydegensac). The additional component compared to the previous algorithm is the local affine frame check constraint proposed in [20] for the fundamental matrix estimation.

GC-RANSAC (author). The implementation provided by the authors. It uses a graph-cut-based local optimization that considers the spatial coherence of the input data points. The additional components are:

1. Sample cheirality check to reject minimal samples early if the implied plane flips between the two views.
2. Single-sided re-projection error, measured in the second image, is used as point-to-model residual.
3. Truncated quadratic cost function and fast iterative local optimization scheme.
4. Gaussian elimination for fast homography estimation from minimal samples.
5. Column-pivoting QR decomposition for larger-than-minimal samples.

GC-RANSAC with PROSAC (author). The previous algorithm with PROSAC sampling [6].

GC-RANSAC (OpenCV) and MAGSAC++ (OpenCV). The OpenCV implementation of the GC-RANSAC and MAGSAC++ algorithms. Additional features:

1. Sequential Probability Ratio Test [7].
2. Gaussian elimination for fast homography estimation.
3. Sample cheirality check to reject minimal samples early if the implied plane flips between the two views.
4. Levenberg-Marquardt numerical optimization minimizes the re-projection error on the final set of inliers.

RHO (OpenCV). The OpenCV implementation of the method proposed in [3]. The components are:

1. PROSAC sampling [6].
2. Sequential Probability Ratio Test [7].
3. Gaussian elimination for fast homography estimation.
4. Sample cheirality check to reject minimal samples early if the implied plane flips between the two views.

MAGSAC (author) and MAGSAC++ (author). The implementations provided by the authors. They use the following additional components for homography estimation.

1. PROSAC sampling [6].
2. Gaussian elimination for fast homography estimation from minimal samples.
3. Column-pivoting QR decomposition for larger-than-minimal samples.
4. Sample cheirality check to reject minimal samples early if the implied plane flips between the two views.

VSAC (author). The implementation provided by the authors. They use the following additional components for homography estimation.

1. Gaussian elimination for fast homography estimation from minimal samples.
2. Householder QR decomposition for larger-than-minimal samples.
3. Sample cheirality check to reject minimal samples early if the implied plane flips between the two views.
4. Local optimization: non-minimal estimation on small subset of inliers (around 15-20 iterations)
5. The MAGSAC++ optimization is applied in the end.

VSAC with PROSAC (author). The previous algorithm using PROSAC sampling [6].

EAS (author). The implementation provided by the authors for the recently proposed algorithm in [10]. The method is implemented in pure Python using the numpy [14] package.

Affine-RANSAC (author). The implementation provided by the authors for the method in [2] using affine correspondences to estimate the homography. Because our benchmark does not have affine correspondences, we approximate them using the SIFT features. Given rotations $\alpha_1, \alpha_2 \in [0, 2\pi]$ and scales s_1, s_2 in the two images for a correspondence, the affine transformation is calculated as $\mathbf{A} = \mathbf{J}_2\mathbf{J}_1^{-1}$, where $\mathbf{J}_i = \mathbf{R}_i\mathbf{S}_i$, $\mathbf{R}_i$ is the 2D rotation matrix rotating by α_i degrees, and $\mathbf{S}_i$ is the 2D scale matrix uniformly scaling

by s_i, $i \in [1, 2]$, along each axis. They use the following additional components for homography estimation.

1. SVD decomposition for estimating the homography affine correspondences.
2. Sample cheirality check adapted for affine correspondences to reject minimal samples early.
3. Graph-Cut RANSAC is used as robust estimator exploiting affine correspondences.

2.2. Deep Pre-filtering

The standard two-view matching pipeline with SIFT or other local features uses SNN ratio test [19] to filter-out unreliable correspondences before running RANSAC, otherwise, the inlier ratio is too small to have good results [4, 9, 17]. Recently, it was shown [4, 21, 24, 26, 28, 29] that using neural networks for correspondence pre-filtering might provide significant benefits over the SNN ratio.

We evaluated how using such models for correspondence pre-filtering for uncalibrated epipolar geometry help homography estimation algorithms. For our study, we took pre-trained models, provided by the authors of each paper and use them for scoring the correspondences. We emphasize that we have neither trained, nor fine-tuned them for the homography estimation task, so their performance is sub-optimal compared to the same models, but trained for the homography estimation. The reason why we did not take the pre-trained models for homography is that authors do not provide them. Since the sought homographies represent 3D planes in the COLMAP reconstruction, they stem from static structures. The homography is thus consistent with the epipolar geometry of the static background. Thus, filtering the correspondences with deep networks trained on epipolar geometry estimation reduces the outlier ratio also for homographies and makes the robust estimation easier. Unless stated otherwise, all the pre-trained models we used, were trained on subset [26] of YCC100M dataset correspondences for fundamental matrix estimation.

CNe [26]. Context normalization networks (CNe) is the first paper on the topic which proposed to use PointNet (MLP) with batch normalization [16] as "context" mechanism. The model does not use any side information and the input is just a set of pair of coordinates in both images.

ACNe [24]. Attentive context normalization networks introduces a special architectural block for the task. The model does not use any side information.

DFE [21]. Deep Fundamental matrix estimation uses differentiable iterative re-weighted least squares for the epipolar geometry estimation and the model predicts weights. It uses the following side information in addition to the point coordinates: difference in scale and orientation of the SIFT features, SNN ratio score, absolute descriptor difference score. Different from the rest of models, DFE was trained on Tanks and Temples dataset, which is smaller and less diverse in terms of camera poses than YCC100M dataset.

OANet [28]. The OANet algorithm introduced several architectural blocks for the correspondence filtering estimation. It also uses the SNN ratio value and mutual nearest neighbor check as a side information.

Neural guiding [4]. Neural-guided RANSAC paper uses a CNe-like architecture, but different training objective (reinforcement learning) and way of utilizing correspondence scores – to perform importance sampling in the RANSAC. Note that we do not use the full NG-RANSAC as proposed in the paper, because there is no author implementation of it – only the fundamental and essential matrix estimation. Instead, we only use the pre-trained model that scores the correspondences. It uses SNN ratio as a side information.

CLNet [29]. CLNet introduces algorithmic and architectural advancement to first remove gross outliers but iterative pruning and only then look for the inlier candidates. No side information is used.

AdaLAM [5]. AdaLAM is handcrafted correspondence filtering method based on local affine transform estimation and scale/orientation consistency check.

3. Uncertainty of Keypoints

The evaluation aims to determine bias and variance of angular, scale, and positional transformations of detected correspondences of SIFT keypoints $\{C_i\}_{i=1}^{N}$. Such statistics calculated on the same dataset allow comparison of different implementations of SIFT detectors. Moreover, we can compare the uncertainty of keypoints orientation, scale, and positions for any detector if such measurements are provided. We followed [1] to derive an affine transformation (4 DoF) $\widetilde{\mathbf{A}}_i \in \mathbb{R}^{2\times 2}$ in the vicinity of the keypoint pair from the reference homography $\widetilde{\mathbf{H}}_i$. The following sections discuss the evaluation of the positional differences, the determination of the reference scale ratios $\widetilde{r}_i$ and of the reference rotations $\widetilde{\alpha}_i$ and the transformation errors in detail. All the evaluations are measured on the OpenCV implementation of the SIFT detector.

3.1. The positional transformation uncertainty

The symmetric positional residual of each keypoint pair depends on the mean reprojection error

$$\epsilon_{x_i} = \sqrt{(|\mathbf{x}'_i - \widetilde{\mathcal{H}}(\mathbf{x}_i)|_2^2 + |\mathbf{x}_i - \widetilde{\mathcal{H}}^{-1}(\mathbf{x}'_i)|_2^2)/8}. \quad (1)$$

The histogram of residuals ϵ_{x_i} of 6.1M keypoint pairs is in Figure 1. Furthermore, the authors W. Förstner and B. P. Worbel [13] show that the standard deviation of the

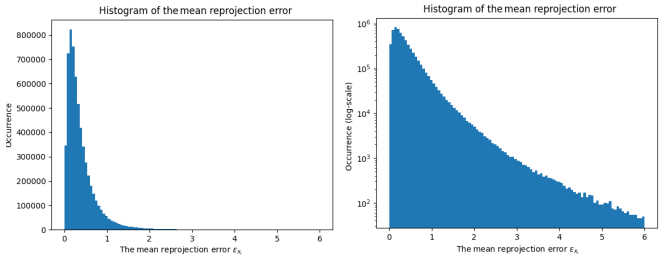

Figure 1. The residuals ϵ_{x_i} of 6.1M keypoint pairs. The right histogram shows the logarithmic scale of the occurrence to visualize the distribution of the residuals. Measured standard deviation $\hat{\sigma}_x \approx 0.67$ pixels. The STD is a factor two larger, than expected, which might result from accepting small outliers.

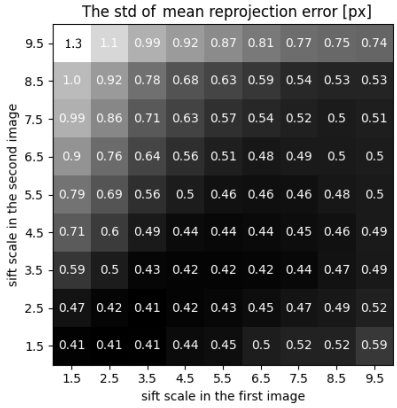

Figure 2. The standard deviation of ϵ_{x_i} for individual scale s_i,s'_i combinations. We can see the dependence of reprojection accuracy on the scale of the related keypoints.

keypoint depends on the detector scale (see [13] p.681, [27] Eq.(15)). Therefore, it is reasonable to assume that the positional transformation error ϵ_{x_i} also depends on keypoint scales s_i, s'_i. We clustered the symmetric positional residuals w.r.t. related s_i, s'_i scales and measured the standard deviation for individual bins, see Figure 2.

3.2. The scale transformation uncertainty

The scale transformation uncertainty is evaluated using the ratios $r_i = s'_i/s_i$ (not to be confused with the redundancy numbers in the main paper) with the scales (s_i, s'_i) from the SIFT keypoints. The scale transformation accuracy is based on the ratio $\Delta r := r_i/\tilde{r}_i$, where the ground truth scale ratio is derived from $\tilde{\mathbf{A}}$ via

$$\tilde{r}_i = \sqrt{|\widetilde{\mathbf{A}_i}|}. \tag{2}$$

For the cases with the affinity matrix having a condition number > 1.5, the shears are assumed to have a too large impact on the scales. We only analyze cases with small scale ratios, *i.e.*, assume values $\tilde{r}_i \in [0.5, 2]$. This interval contains 99.62% keypoint pairs. Further, the weighted

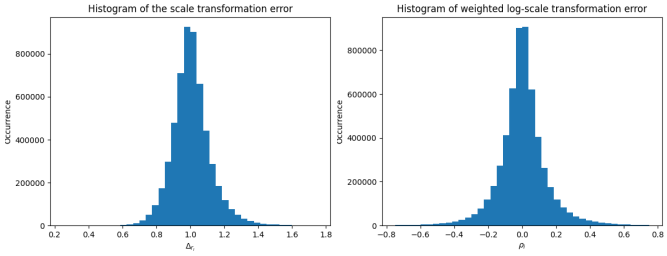

Figure 3. The histogram of the scale transformation ratio Δr_i and the weighted log-ratio ρ_i on 5.6M keypoint pairs.

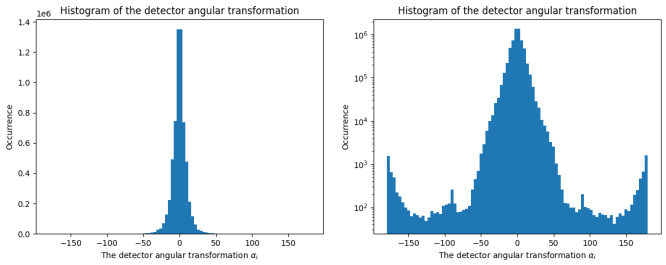

Figure 4. The histogram of the detector angular transformation α_i for 6.1M of keypoint pairs. The right histogram shows logarithmic scale of the occurrence to visualize the number of samples across the complete interval $[-180, 180)$ degrees.

log-ratio $\rho_i = \log(\Delta r_i)/\tilde{r}_i$ is calculated using the filtered Δr_i related to the ground truth $\tilde{r}_i$. The scale statistics of the remaining 5.6M keypoint pairs are shown in Figure 3.

3.3. The angular transformation uncertainty

The histogram of angular transformation $\alpha_i = \phi'_i - \phi_i$ for all keypoint pairs is visualized in Figure 4. The uncertainty of this transformation can be calculated by: (1) comparing direction vectors $\mathbf{d}(\phi'_i)$ with the transformed direction $\mathbf{d}(\phi_i)$ into the coordinates of $\mathbf{d}(\phi'_i)$ or (2) deriving a local rotation from the reference homography and comparing it to the keypoint angular transformation.

3.3.1 Comparing direction vectors

The directional vector $\mathbf{d}_i = [\cos(\phi_i) \;\; \sin(\phi_i)]^{\mathrm{T}}$ realizing the first keypoint orientation can be transformed into the second image by the multiplication with the local approximation of affinity transformation (4DoF)

$$\bar{\mathbf{d}}_i = \widetilde{\mathbf{A}}_i \mathbf{d}_i. \tag{3}$$

The multiplication with the local affinity $\widetilde{\mathbf{A}}_i \in \mathbb{R}^{2\times 2}$ does not include the projective part. The angle in the interval $[-\pi, \pi]$ can be obtained by

$$\Delta\alpha_{\text{direct}_i} = \angle(\mathbf{d}'_i, \bar{\mathbf{d}}_i) = \text{atan2}(|[\mathbf{d}'_i, \bar{\mathbf{d}}_i]|, \mathbf{d}'^T_i \bar{\mathbf{d}}_i). \tag{4}$$

This is a reasonable measure for evaluating the quality of the directions since – assuming no outliers – the expected

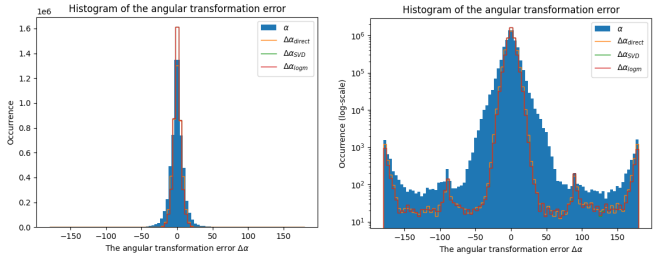

Figure 5. The histogram of angular transformation error $\Delta\alpha$ on top of α. The transformation was (1.) measured as the angle between directional vectors, $\Delta\alpha_{\text{direct}}$ eq. (4) , (2.) subtracting the reference angular transformation decomposed by SVD, $\Delta\alpha_{\text{SVD}}$ eq. (10), and (3.) subtracting the ground truth angular transformation obtained from the exponential analysis, $\Delta\alpha_{\text{logm}}$ eq. (16). We assumed 4.3M correspondences with $\text{cond}(\widetilde{\mathbf{A}}_i) < 1.2$. The standard deviation is $\hat{\sigma}_\alpha \approx 7.9^\circ$ is approximately two times the rounding error.

value of this angular difference is zero,[1] $\text{E}(\underline{\Delta\alpha}_{\text{direct}_i}) = 0$. Fig. 5 shows the histogram of the $\Delta\alpha_{\text{direct}_i}$ from (4).

3.3.2 Partitioning of an affinity

We assume $\widetilde{\mathbf{A}}_i \in \mathbb{R}^{2\times 2}$ matrix locally approximate the homography $\widetilde{\mathbf{H}}_i \in \mathbb{R}^{3\times 3}$. The goal of comparing SIFT directions could be to determine the rotation component $\tilde{\mathbf{R}}$ of the affinity $\widetilde{\mathbf{A}}_i$ and compare it to the angle between the directions of corresponding keypoints.

We address three alternatives for determining the rotational component of $\tilde{\mathbf{A}}$:

1. a QR-decomposition,
2. a SVD-decomposition, and
3. an exponential decomposition.

Rotation from QR-decomposition of an affinity A. Assuming the affinity is a concatenation of a shear matrix $\mathbf{S}$ and a subsequent rotation with $\mathbf{R}$

$$\mathbf{A} = \mathbf{RS} \tag{5}$$

the classical QR-decomposition is defined as

$$\mathbf{R}_{\text{qr,A}} := \mathbf{R} \quad \text{with} \quad [\mathbf{R}, \mathbf{S}] = \text{qr}(\mathbf{A})\,. \tag{6}$$

In case the affinity is defined by the reverse sequence, *i.e.*

$$\mathbf{A} = \mathbf{SR} \tag{7}$$

the QR decomposition of the transposed needs to be taken

$$\mathbf{R}_{\text{qr,A}^\mathsf{T}} := \mathbf{R}^\mathsf{T} \quad \text{with} \quad [\mathbf{R}, \mathbf{S}] = \text{qr}(\mathbf{A}^\mathsf{T})\,. \tag{8}$$

If there are no shears, *i.e.* the shear matrix is a scaled unit matrix, the two rotations $\mathbf{R}_{\text{qr,A}}$ and $\mathbf{R}_{\text{qr,A}^\mathsf{T}}$ are the same, otherwise they differ.

[1] Stochastical variables are underscored

Rotation from SVD-decomposition of A. An alternative way to derive the rotation component uses the matrix exponential. Let us assume, the affinity is decomposable as two rotations sandwiching a individual scaling

$$\mathbf{A} = \mathbf{UDV}^\mathsf{T} \quad \text{with} \quad \mathbf{D} = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix}, \tag{9}$$

where the shears are represented by the rotation $\mathbf{V}$ and the ratio d_1/d_2. Then the SVD yields the rotation

$$\mathbf{R}_{\text{svd,A}} := \mathbf{UV}^\mathsf{T} \quad \text{with} \quad [\mathbf{U}, \mathbf{\Lambda}, \mathbf{V}] = \text{svd}(\mathbf{A})\,. \tag{10}$$

Transposing $\mathbf{A}$ does not change the rotation. The resulting rotation only is identical to those of the QR-decomposition if the affinity is a scaled rotation.

Rotation from an exponential decomposition The affinity $\mathbf{A}$ can be written as an exponential of a matrix $\mathbf{B}$

$$\mathbf{A} = \text{e}^{\mathbf{B}} \tag{11}$$

If the matrix $\mathbf{B}$ is zero, i.e. $\mathbf{B} = \mathbf{0}$, the affinity is a unit transformation. We now can decompose the exponent additively in the following form

$$\mathbf{B} = \sum_i p_i \mathbf{B}_i \tag{12}$$

with the four basic 2×2 matrices

$$\mathbf{B}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{B}_2 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \tag{13}$$

$$\mathbf{B}_3 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{B}_4 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{14}$$

Hence

$$\mathbf{A} = \text{e}^{p_1\mathbf{B}_1 + p_2\mathbf{B}_2 + p_3\mathbf{B}_3 + p_4\mathbf{B}_4}\,. \tag{15}$$

If we take each of the summands individually, the four parameters refer to (1) scaling with $\log p_1$, (2) rotation by p_2 [rad], (4) 1st shear, namely opposite scaling of axes, and (4) 2nd shear, namely opposite rotation of axes. The rotation is given by the well known relation

$$\mathbf{R} = \exp(p_2\mathbf{B}_2)\,. \tag{16}$$

Furthermore, for the first shear we explicitely have

$$\exp\left(\begin{bmatrix} 0 & p_4 \\ p_4 & 0 \end{bmatrix}\right) \tag{17}$$

$$= \begin{bmatrix} \text{e}^{-p_4/2} + \text{e}^{p_4/2} & \text{e}^{p_4/2} - \text{e}^{-p_4/2} \\ \text{e}^{-p_4/2} - \text{e}^{-p_4/2} & \text{e}^{-p_4/2} + \text{e}^{p_4/2} \end{bmatrix} \tag{18}$$

$$\overset{q_4=\text{e}^{p_4/2}}{=} \begin{bmatrix} q_4 + 1/q_4 & q_4 - 1/q_4 \\ q_4 - 1/q_4 & q_4 + 1/q_4 \end{bmatrix}. \tag{19}$$

This representation is highly symmetric. The additive terms are invariant w.r.t. the sequence of the terms. Moreover, the scaled rotation is independent on the existence of shears.

However, since the exponent of two matrices only is the product of the two matrices if they commute, i.e.

$$\exp(\mathbf{A}+\mathbf{B}) = \exp(\mathbf{A})\exp(\mathbf{B}) \quad \text{only if} \quad \mathbf{AB}=\mathbf{BA}\,, \tag{20}$$

the interpretation of the elements in the exponent is not independent of the existence of the other elements. Only a common scaling can be exchanged with the other components, as is known from scaled rotation.

Now, we can define the rotational component using (16) deriving p_2 from

$$p_2 = (B(2,1) - B(1,2))/2 \quad \text{with} \quad \mathbf{B} = \log(\mathbf{A}) \tag{21}$$

where $\log(\mathbf{A})$ is the matrix logarithm of $\mathbf{A}$.

Therefore we are able to identify the existence of shears, namely we have no shears if

$$d_s^2 = |[p_3, p_4]| = p_3^2 + p_4^2 = 0 \tag{22}$$

Since a scale rotation has condition number $\text{cond}(s\mathbf{R}) = 1$, also the condition number can be used to identify the lack of shears, namely if $\text{cond}(\mathbf{A}) = 1$. For not too large shears the the condition number and the degree of shears d_s^2 are approximately the same:

$$d_s^2 \approx \text{cond}(\mathbf{A})\,. \tag{23}$$

4. Effect of Weighting and Estimation Type

4.1. Outline of the analysis

We use two sets of sample data to answer two questions:

1. What loss in accuracy is to be expected when using an algebraic estimation vs. a ML-estimation?

2. What effect on the accuracy does a scale dependent weighting have onto the results of an ML-estimation? (see [13], Sect. 15.4.1.3)

The first set A was chosen, such that (1) the number of correspondences is small in order to allow for non-uniform distribution of points and (2) the shears to be large, the planes are not fronto-parallel in order to have the homographies largely deviate from a scaled rotation. The second set B is the same as been used for the investigation into the uncertainty of the SIFT detector.

4.2. Algebraic and ML minimization

We apply two estimation methods, each yielding covariance matrices for the homography parameters based on the constraints using the observations, containing the homogeneous coordinates of the keypoint pairs $\mathbf{l}_i := [\mathbf{x}_i, \mathbf{x}_i']^T$ with the covariance matrix $\mathbf{\Sigma}_{l_i l_i}$ and the unknown parameters $\boldsymbol{\theta} := \text{vec}\mathbf{H}$ and $\mathbf{y}_i = \mathbb{E}(\mathbf{l}_i)$

$$\mathbf{0} = \mathbf{g}_i(\boldsymbol{\theta}, \mathbf{y}_i)) := \mathbb{E}(\mathbf{x}_i'^T) \times (\mathbf{H}\,\mathbb{E}(\mathbf{x}_i)) \tag{24}$$

which linearized has the form

$$\mathbf{g}_i(\boldsymbol{\theta}, \mathbf{y}_i) = \mathbf{g}_i(\boldsymbol{\theta}^0, \mathbf{y}_i^0) + \mathbf{A}_i\Delta\boldsymbol{\theta} + \mathbf{B}_i^T\Delta\mathbf{y}_i = \mathbf{0}\,. \tag{25}$$

with the Jacobians

$$\mathbf{A}_i = \frac{\partial \mathbf{g}_i}{\partial \boldsymbol{\theta}} \quad \text{and} \quad \mathbf{B}_i^T = \frac{\partial \mathbf{g}_i}{\partial \mathbf{l}_i}\,, \tag{26}$$

yielding the complete observation vector, constraints, and complete Jacobians

$$\mathbf{l} = [\mathbf{l}_i]\,, \quad \mathbf{g} = [\mathbf{g}_i]\,, \quad \mathbf{A} = [\mathbf{A}_i]\,, \quad \text{and} \quad \mathbf{B} = \text{Diag}(\mathbf{B}_i^T)\,,$$

i.e. using the block diagonal matrix Diag(·) with the $\mathbf{B}_i^T$ as entries. We obtain the following covariance matrices for the homography parameters.

1. The *classical algebraic method* minimizing the algebraic error

   $$\Omega^{(\text{ALG})}(\boldsymbol{\theta}) = \mathbf{g}^T(\boldsymbol{\theta}, \mathbf{l})\mathbf{g}(\boldsymbol{\theta}, \mathbf{l})\,, \tag{27}$$

   yields the linear relation from (25)

   $$\widehat{\Delta\boldsymbol{\theta}} = -(\mathbf{A}^T\mathbf{A})^+\mathbf{A}^T\mathbf{B}^T\Delta\mathbf{l}\,, \tag{28}$$

   see [13], eq. (4.518), from which we obtain the covariance matrix

   $$\mathbf{\Sigma}_{\hat\theta\hat\theta}^{(\text{ALG})} = (\mathbf{A}^T\mathbf{A})^+\mathbf{A}^T\mathbf{B}^T\mathbf{\Sigma}_{ll}\mathbf{B}\mathbf{A}(\mathbf{A}^T\mathbf{A})^+\,. \tag{29}$$

   Observe, we only would obtain the covariance matrix $(\mathbf{A}^T\mathbf{A})^{-1}$ if the covariance matrix $\mathbf{B}^T\mathbf{\Sigma}_{ll}\mathbf{B}$ of the constraints $\mathbf{A}\Delta\boldsymbol{\theta}$ would be the unit matrix, see (25), which generally does not hold.

2. The *ML-estimation*, taking the uncertainty of the points into account, minimizes

   $$\Omega^{(\text{ML})}(\boldsymbol{\theta}) = \mathbf{v}^T\mathbf{\Sigma}_{ll}^{-1}\mathbf{v} \tag{30}$$

   with $\mathbf{v} = \mathbf{y} - \mathbf{l} = (\mathbf{y}^0 + \Delta\mathbf{y}) - \mathbf{l}$, under the constraints (25), which include the unknown parameters $\boldsymbol{\theta}$, and yields the linear relation

   $$\widehat{\Delta\boldsymbol{\theta}} = -\mathbf{\Sigma}_{\hat\theta\hat\theta}^{(\text{ML})}\mathbf{A}^T(\mathbf{B}^T\mathbf{\Sigma}_{ll}\mathbf{B})^{-1}\mathbf{B}^T\Delta\mathbf{l}\,, \tag{31}$$

   see [13], eq. (4.447) with the covariance matrix

   $$\mathbf{\Sigma}_{\hat\theta\hat\theta}^{(\text{ML})} = (\mathbf{A}^T(\mathbf{B}^T\mathbf{\Sigma}_{ll}\mathbf{B})^{-1}\mathbf{A})^+\,. \tag{32}$$

   If we assume $\mathbf{B}^T\mathbf{\Sigma}_{ll}\mathbf{B} = \mathbf{I}$, which is what algebraic minimization does, eq. (31) reduces to obtain (28).

In both cases we do not make the procedural details explicit, which are caused by the redundant representation of the homography and the homogeneous coordinates: Actually, the covariance matrix $\mathbf{\Sigma}_{\hat{\theta}\hat{\theta}}$ has rank 8, since the homography only has 8 d.o.f., similarly, the covariance matrix of a homogeneous vector $\mathbf{x}$ representing a 2D point, is rank 2. In both cases, we employ a minimal representation in the tangent space defined by the constraints $||\mathbf{H}||_2 = 1$ and $|\mathbf{x}|_2 = 1$. Details for an ML-estimation of a homography are given in [13], Sect. 10.6.3.

4.3. Scale dependent weighting

We use two different weighting schemes for the ML-estimation

1. Equal weights for all points
$$w_1(i) = w_0(i') = 1\,. \tag{33}$$
2. Choosing the weights as a function of the scales of the I keypoints, namely
$$w_s(i) = \frac{m^2}{s^2(i)} \quad \text{and} \quad w_s(i') = \frac{m^2}{s^2(i')}\,, \tag{34}$$
with the geometric mean of all scales
$$m = \left(\prod_i s_i \prod_{i'} s_{i'}\right)^{1/(2I)}\,. \tag{35}$$
The denominator is meant to have the average variance 1, to be comparable to (33), though the results do not depend on this common scaling.

For the ML-estimates, in addition to the covariance matrices $\mathbf{\Sigma}_{\hat{\theta}\hat{\theta}}$ we also obtain the estimated variance factor
$$\hat{\sigma}_0^2 = \frac{\Omega(\hat{\boldsymbol{\theta}})}{R} \tag{36}$$
which depends on the weighted sum Ω of the squared residuals, i.e. the reprojection errors and the redundancy $R = 2I - 8$ of the estimation. It tells by which factor we need to multiply the assumed covariance matrix in order to obtain an unbiased covariance matrix, assuming the given covariance matrix provides the correct ratio of the uncertainties between the observations:
$$\mathbf{\Sigma}_{\hat{\theta}\hat{\theta}}^{\text{a posteriori}} = \sigma_0^2 \mathbf{\Sigma}_{\hat{\theta}\hat{\theta}}^{\text{a priori}}\,. \tag{37}$$

4.4. Accuracy Evaluation criteria

We use the following *criteria* to determine the loss in accuracy, i.e. an increase of the standard deviations $\sigma_{\hat{\theta}_u}$, when comparing the covariance matrix $\mathbf{\Sigma}$ to a reference covariance matrix $\widetilde{\mathbf{\Sigma}}$, namely the mean loss
$$l_{\text{mean}} = \sqrt{\text{tr}(\mathbf{\Sigma}_{\hat{\theta}\hat{\theta}}\widetilde{\mathbf{\Sigma}}_{\hat{\theta}\hat{\theta}}^{-1})/8} \tag{38}$$
and the maximum loss
$$l_{\text{max}} = \sqrt{\max \lambda(\mathbf{\Sigma}_{\hat{\theta}\hat{\theta}}\widetilde{\mathbf{\Sigma}}_{\hat{\theta}\hat{\theta}}^{-1})}\,, \tag{39}$$
see [12]. In case the two matrices are diagonal matrices with the variances, we obtain *the average and the maximum ratio of the standard deviations*.

5. Geometry and Statistics for Sect. 5

5.1. On the estimate ϵ_{x_i} for σ_i

We show, that
$$\epsilon_{x_i} = \sqrt{(|\mathbf{x}_i - \mathcal{H}_i(\mathbf{x}_i')|_2^2 + |\mathbf{x}_i' - \mathcal{H}_i^{-1}(\mathbf{x}_i)|_2^2)/8} \tag{40}$$
is a meaningful estimate for the standard deviation σ_i of all coordinates u_{ij} and u_{ij}' of the given points $\mathbf{x}_i = (u_{i1}, u_{i2})$ and $\mathbf{x}_i' = (u_{i1}', u_{i2}')$. Hence, we assumeD$(\underline{\mathbf{e}}_i) = \text{E}(\underline{\mathbf{e}}_i\underline{\mathbf{e}}_i^\mathsf{T}) = \text{D}(\underline{\mathbf{e}}_i') = \text{E}(\underline{\mathbf{e}}_i'\underline{\mathbf{e}}_i'^\mathsf{T}) = \sigma_i^2\mathbf{I}_2$, which holds for the errors $\underline{\mathbf{e}}_i = \underline{\mathbf{u}}_i - \text{E}(\underline{\mathbf{u}}_i)$ and $\underline{\mathbf{e}}_i' = \underline{\mathbf{u}}_i' - \text{E}(\underline{\mathbf{u}}_i')$. Linearizing $\mathbf{x}_i - \mathcal{H}_i(\mathbf{x}_i')$ leads to $\underline{\mathbf{e}}_i - \mathbf{A}_i(\underline{\mathbf{e}}_i')$, and similarly for the second term. Thus, the RMSE, *i.e.* the expession under the squareroot in (40) is linearized to
$$\Omega_i = |\mathbf{e}_i - \mathbf{A}_i(\mathbf{e}_i')|_2^2 + |\mathbf{e}_i' - \mathbf{A}_i^{-1}(\mathbf{e}_i)|_2^2 \tag{41}$$
We now determine the expectation $\text{E}(\underline{\Omega}_i)$ and obtain
$$\begin{aligned}
\text{E}(\underline{\Omega}_i) &= \text{E}\left((\underline{\mathbf{e}}_i - \mathbf{A}_i(\underline{\mathbf{e}}_i'))^\mathsf{T}(\underline{\mathbf{e}}_i - \mathbf{A}_i(\underline{\mathbf{e}}_i'))\right. && (42)\\
&\quad \left.+(\underline{\mathbf{e}}_i' - \mathbf{A}_i^{-1}(\mathbf{e}_i))^\mathsf{T}(\underline{\mathbf{e}}_i' - \mathbf{A}_i^{-1}(\mathbf{e}_i))\right) && (43)\\
&= \text{E}\left(\underline{\mathbf{e}}_i^\mathsf{T}\underline{\mathbf{e}}_i + \underline{\mathbf{e}}_i'^\mathsf{T}\mathbf{A}_i^\mathsf{T}\mathbf{A}_i\underline{\mathbf{e}}_i'\right. && (44)\\
&\quad \left.+\mathbf{e}_i'^\mathsf{T}\underline{\mathbf{e}}_i' + \underline{\mathbf{e}}_i^\mathsf{T}\mathbf{A}_i^{-T}\mathbf{A}_i^{-1}\underline{\mathbf{e}}_i\right) && (45)
\end{aligned}$$
With $\text{tr}(\mathbf{UV}) = \text{tr}(\mathbf{VU})$, thus $\mathbf{a}^\mathsf{T}\mathbf{S}\mathbf{a} = \text{tr}(\mathbf{a}^\mathsf{T}\mathbf{S}\mathbf{a}) = \text{tr}(\mathbf{S}\mathbf{a}\mathbf{a}^\mathsf{T})$ we then obtain
$$\begin{aligned}
\text{E}(\underline{\Omega}_i) &= \text{E}\left(\underline{\mathbf{e}}_i^\mathsf{T}\underline{\mathbf{e}}_i + \underline{\mathbf{e}}_i'^\mathsf{T}\mathbf{A}_i^\mathsf{T}\mathbf{A}_i\underline{\mathbf{e}}_i'\right. && (46)\\
&\quad \left.+\underline{\mathbf{e}}_i'^\mathsf{T}\underline{\mathbf{e}}_i' + \underline{\mathbf{e}}_i^\mathsf{T}\mathbf{A}_i^{-T}\mathbf{A}_i^{-1}\underline{\mathbf{e}}_i\right) && (47)\\
&= \text{E}\left(\text{tr}(\underline{\mathbf{e}}_i\underline{\mathbf{e}}_i^\mathsf{T}) + \text{tr}(\mathbf{A}_i^\mathsf{T}\mathbf{A}_i\underline{\mathbf{e}}_i'\underline{\mathbf{e}}_i'^\mathsf{T})\right. && (48)\\
&\quad \left.+\text{tr}(\mathbf{e}_i'\mathbf{e}_i'^\mathsf{T}) + \text{tr}(\mathbf{A}_i^{-T}\mathbf{A}_i^{-1}\underline{\mathbf{e}}_i\underline{\mathbf{e}}_i^\mathsf{T})\right) && (49)\\
&= \text{tr}(\text{E}(\underline{\mathbf{e}}_i\underline{\mathbf{e}}_i^\mathsf{T})) + \text{tr}(\mathbf{A}_i^\mathsf{T}\mathbf{A}_i\text{E}(\underline{\mathbf{e}}_i'\underline{\mathbf{e}}_i'^\mathsf{T})) && (50)\\
&\quad +\text{tr}(\text{E}(\underline{\mathbf{e}}_i'\underline{\mathbf{e}}_i'^\mathsf{T})) + \text{tr}(\mathbf{A}_i^{-T}\mathbf{A}_i^{-1}\text{E}(\mathbf{e}_i\mathbf{e}_i^\mathsf{T})) && (51)\\
&= \text{tr}(\mathbf{I}_2)\sigma_i^2 + \text{tr}(\mathbf{A}_i^\mathsf{T}\mathbf{A}_i)\sigma_i^2 && (52)\\
&\quad +\text{tr}(\mathbf{I}_2)\sigma_i^2 + \text{tr}(\mathbf{A}_i^{-\mathsf{T}}\mathbf{A}_i^{-1})\sigma_i^2 && (53)\\
&= (4 + \text{tr}(\mathbf{A}_i^\mathsf{T}\mathbf{A}_i) + \text{tr}(\mathbf{A}_i^{-\mathsf{T}}\mathbf{A}_i^{-1}))\sigma_i^2\,. && (54)
\end{aligned}$$
With the eigenvalues $\lambda_{1,2}(\mathbf{A}_i^\mathsf{T}\mathbf{A}_i)$ we now obtain
$$\text{E}(\underline{\Omega}_i) = (4 + \lambda_1 + \lambda_2 + 1/\lambda_1 + 1/\lambda_2)\sigma_i^2 \geq 8\sigma_i^2 \tag{55}$$

since $x + 1/x = (1-x)^2/x + 2 \geq 2$ for $x > 0$. Hence, if $\lambda_1 = \lambda_2 = 1$, thus for a pure rotation, the value $\epsilon^2_{x_i}$ is an unbiased estimator for σ_i^2. Dividing the RMSE Ω by $\sqrt{8}$ therefore leads to a conservative estimate of the standard deviation σ_i.

5.2. Affinity and Slope of Plane

We give a relation between the condition number and the slope of a plane observed by an image pair in normal position.

The image of a sloped plane leads to scale differences s and shears a due to the tilts Z_x and Z_y of the plane along and across the base line. They have the form

$$\mathbf{A}_s = \begin{bmatrix} 1+s & 0 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{A}_a = \begin{bmatrix} 1 & a \\ 0 & 1 \end{bmatrix} \tag{56}$$

The combined effect is the affinity

$$\mathbf{A}_{sa} = \mathbf{A}_a \mathbf{A}_s = \begin{bmatrix} 1+s & a \\ 0 & 1 \end{bmatrix} \tag{57}$$

Condition Number for Affinity except Scaled Rotation. The condition number of this affinity, is given by

$$c = 1 + \frac{(\sqrt{4(1+s)+t^2}+t)}{2(1+s)}t \quad \text{with} \quad t^2 = a^2 + s^2 \tag{58}$$

For small s and t it can be approximated by

$$c \approx 1 + t = 1 + \sqrt{a^2 + s^2} \tag{59}$$

neglecting higher order terms. A condition number $c = 1.5$ corresponds to $t = \sqrt{2}/3$.

Affine Parameters and Slope of Scene Plane. Assume the stereo image pair in normal position with rotation $\mathbf{R} = \mathbf{I}$, basis $\mathbf{b} = [1,0,0]^\mathsf{T}$, and focal length $f = 1$ with coordinate system in the first camera observing a sloped plane at $[0,0,Z_0]^\mathsf{T}$

$$Z = Z_0 + XZ_X + YZ_Y \quad \text{with} \quad Z_X = \frac{\partial Z}{\partial X}, \quad Z_Y = \frac{\partial Z}{\partial Y} \tag{60}$$

or with homogeneous plane coordinates

$$\mathbf{A} = [Z_X, Z_Y, -1, Z_0]^\mathsf{T} = [\mathbf{n}^\mathsf{T}, Z_0]^\mathsf{T}. \tag{61}$$

The homography from $\mathbf{x}'$ to $\mathbf{x}''$ is given by $\mathbf{x}'' = \mathbf{H}\mathbf{x}'$ by

$$\mathbf{H} = \mathbf{I} + \frac{\mathbf{b}\mathbf{n}^\mathsf{T}}{Z_0} = \begin{bmatrix} \frac{Z_X}{Z_0}+1 & \frac{Z_Y}{Z_0} & -\frac{1}{Z_0} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{62}$$

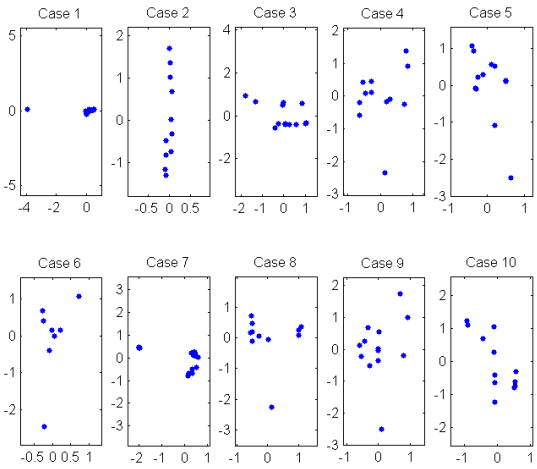

Figure 6. The distribution of the points for the 10 cases, see Tab. 1

This is an affinity with a Jacobian independent of the position in the image, namely

$$\mathbf{A} = \frac{\partial \mathbf{x}''}{\partial \mathbf{x}'} = \begin{pmatrix} 1 + \frac{Z_X}{Z_0} & \frac{Z_Y}{Z_0} \\ 0 & 1 \end{pmatrix}. \tag{63}$$

Hence, we have the scale difference and the shear

$$s = \frac{Z_X}{Z_0} \quad \text{and} \quad a = \frac{Z_Y}{Z_0}. \tag{64}$$

If the scale difference and the shear are $s = a = 1/3$, the slope of the plane is $\operatorname{atan}(\sqrt{2}/3) = 25.24°$ and the condition number is $c = 1.5$, which we use as threshold for approximate scale rotations.

6. Results

6.1. Data set A

For the 10 cases of data set A the results are collected in Table 1. Fig. 6 shows the point distribution of the 10 cases.

Discussion. The table allows the following conclusions:

- The prior scale dependent standard deviations lies in a range between 0.24 pixel and 6.11 pixel. Since the redundancy in all cases is small, these values are quite uncertain. However, their ratios, which only depend on the scales of the points, are as uncertain as the scales are. The ratios vary between 4 in case 3, and 23 in case 5.
- Starting from an uncertainty of 1 pixel, the estimated (square rooted) variance factors $\hat{\sigma}_0$ indicate that on an

case	I	$\min(\sigma_x)$ [px]	$\max(\sigma_x)$ [px]	$\frac{\max(\sigma_x)}{\min(\sigma_x)}$	$\sigma_0(w=1)$ [px]	$\sigma_0(w(s))$ [px]	$l^{\mathrm{ALG\|ML}}_{\mathrm{mean}}$	$l^{\mathrm{ALG\|ML}}_{\mathrm{max}}$	$l^{\mathrm{1\|s}}_{\mathrm{mean}}$	$l^{\mathrm{1\|s}}_{\mathrm{max}}$
	1	2	3	4	5	6	7	8	9	10
1	18	0.28	2.65	10	**0.340**	**0.377**	1.005	1.047	1.105	1.350
2	12	0.32	2.52	8	0.170	0.179	1.052	1.318	1.223	1.569
3	13	**0.48**	2.08	4	0.133	0.129	1.050	1.273	1.076	1.251
4	14	0.29	4.66	16	0.266	0.172	**1.105**	**1.418**	**2.343**	4.098
5	13	0.26	**6.11**	**23**	0.228	0.276	1.043	1.218	2.206	**4.157**
6	10	0.27	2.14	8	0.063	0.060	1.037	1.163	1.126	1.348
7	**24**	0.26	2.59	10	0.252	0.287	1.002	1.009	1.131	1.280
8	12	0.24	5.20	22	0.165	0.242	1.023	1.087	2.119	2.989
9	16	0.29	5.62	20	0.261	0.285	1.102	1.331	1.976	3.382
10	14	0.24	5.36	22	0.203	0.296	1.054	1.264	1.860	2.972
mean		0.29	3.89	14	0.208	0.230	1.047	1.213	1.616	2.440
max	24	0.48	6.11	23	0.340	0.377	1.105	1.418	2.343	4.157

Table 1. Data set A. Comparing the accuracy of different homography estimates. Columns: (1) Number I of point pairs, (2–4) minimal and maximal standard deviations $\min(\sigma_x)$ and $\max(\sigma_x)$ of weighted points and their ratio, (5–6) estimated variance factors for unweighted and weighted points, (7–8) mean and maximal losses when comparing the algebraic with the ML-estimate using equal weights, $l^{\mathrm{ALG|ML}}_{\mathrm{mean}}$ and $l^{\mathrm{ALG|ML}}_{\mathrm{max}}$, and (9–10) mean and maximal losses when comparing the unweighted and the weighted ML-estimate, $l^{\mathrm{1|s}}_{\mathrm{mean}}$ and $l^{\mathrm{1|s}}_{\mathrm{max}}$, see Fig. 6

average the keypoint coordinates are much better than 1 pixel, approximately by a factor 4 to 5.

- The loss in accuracy when using the classical algebraic method for homography estimation compared to the achievable accuracy using a ML-estimation is shown in columns 8 and 9. The mean loss mostly is below 10%, which appears acceptable. However, the maximum loss is about 42 % (case 4).

- The loss in accuracy when using equally weighted coordinates instead of taking the (assumed) scale dependency into account is shown in columns 10 and 11. While the mean loss lies between 8% in case 3 and a factor 2.3 in case 4, the maximum loss reaches a factor 4.2 in case 5. The variation of the standard deviations (column 5) is approximately coherent with the loss in accuracy.

6.2. Data set B

Data set B consists of 969 homographies with alltogether 22 489 correspondences. We first provide the result of the first 30 cases in Tab. 2 together with the mean and maximum values for each criterion. The results confirm the findings of data set A, of course leading to more extreme ranges/maximum values.

Analysing the complete data set B with 969 cases yields the results shown in the histograms of Fig 7 and 8. The maximum ratios of the scale dependent standard deviations

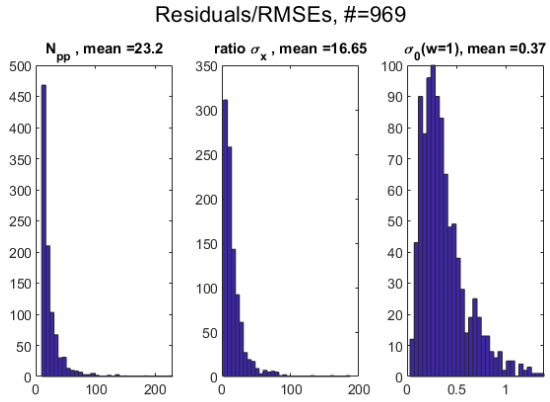

Figure 7. Data set B with 969 cases. Histograms of number of correspondences, ratio of scale dependent standard deviations of coordinates, and estimated (square rooted) variance factor $\hat{\sigma}_0$

on an average are 15.5, which appears to be quite large, however, confirming the results shown in Tab. 2. The mean variance factor is $\overline{\sigma_0^2} = (0.37\ [\mathrm{px}])^2$, being consistent with earlier investigations.

Finally, we determined the *average residuals* for each case using the following symmetric root mean square error .

$$\mathrm{RMSE} = \sqrt{\frac{1}{8I}\sum_{i=1}^{I} |\mathbf{x}'_i - \mathcal{H}(\mathbf{x}_i)|^2 + |\mathbf{x}_i - \mathcal{H}^{-1}(\mathbf{x}'_i)|^2}\,, \tag{65}$$

case	I	$\min(\sigma_x)$ [px]	$\max(\sigma_x)$ [px]	$\frac{\max(\sigma_x)}{\min(\sigma_x)}$	$\sigma_0(w=1)$ [px]	$\sigma_0(w(s))$ [px]	$l_{\text{mean}}^{\text{ALG\|ML}}$	$l_{\text{max}}^{\text{ALG\|ML}}$	$l_{\text{mean}}^{\text{1\|s}}$	$l_{\text{max}}^{\text{1\|s}}$
	1	2	3	4	5	6	7	8	9	10
1	39	0.34	2.18	6	0.216	0.209	1.050	1.192	1.119	1.303
2	44	0.31	6.96	23	0.341	0.358	1.042	1.131	2.511	4.451
3	34	0.37	5.07	14	0.337	0.302	1.146	1.619	1.688	2.267
4	33	0.17	1.82	10	0.238	0.287	1.080	1.510	1.054	1.111
5	17	0.16	4.83	31	0.395	0.460	1.038	1.090	2.112	3.086
6	31	0.40	9.13	23	0.701	0.781	**1.169**	**1.918**	2.084	4.254
7	21	0.39	2.63	7	0.415	0.474	1.112	1.598	1.254	1.770
8	22	0.45	2.60	6	0.255	0.270	1.061	1.223	1.202	1.458
9	16	0.38	7.71	20	0.367	0.494	1.049	1.138	2.274	3.921
10	37	0.34	3.46	10	0.178	0.197	1.052	1.276	1.392	1.770
11	14	0.33	3.54	11	0.097	0.117	1.045	1.308	1.062	1.214
12	30	0.51	1.91	4	0.158	0.161	1.061	1.224	1.087	1.209
13	36	0.24	8.76	37	0.334	0.325	1.007	1.030	3.255	5.093
14	26	0.36	4.14	12	0.302	0.296	1.016	1.052	1.828	2.487
15	42	0.19	**14.79**	**77**	0.542	0.676	1.022	1.082	**4.118**	**7.477**
16	17	0.37	2.65	7	0.234	0.276	1.031	1.117	1.289	1.711
17	22	0.20	1.88	9	0.276	0.318	1.075	1.476	1.099	1.246
18	16	0.23	2.75	12	0.567	0.576	1.005	1.017	1.429	2.252
19	56	0.28	4.50	16	0.478	0.719	1.079	1.300	1.927	2.484
20	48	0.28	3.08	11	0.368	0.463	1.063	1.232	1.884	2.105
21	12	**0.51**	2.93	6	0.117	0.150	1.015	1.072	1.242	2.197
22	**60**	0.19	2.09	11	**0.851**	0.908	1.105	1.655	1.085	1.194
23	18	0.26	1.68	6	0.392	0.464	1.004	1.030	1.150	1.453
24	30	0.33	8.16	25	0.405	0.373	1.103	1.368	2.483	4.903
25	10	0.32	4.99	16	0.398	0.391	1.011	1.054	1.182	1.924
26	14	0.43	7.54	18	0.303	0.368	1.009	1.054	1.526	3.012
27	17	0.17	2.58	15	0.160	0.200	1.058	1.285	1.263	1.593
28	13	0.24	5.89	25	0.292	0.331	1.027	1.175	1.621	2.723
29	11	0.26	5.37	20	0.548	0.580	1.003	1.020	1.380	1.989
30	20	0.28	9.41	34	0.700	**0.980**	1.030	1.126	2.474	4.064
case	I	$\min(\sigma_x)$ [px]	$\max(\sigma_x)$ [px]	$\frac{\max(\sigma_x)}{\min(\sigma_x)}$	$\sigma_0(w=1)$ [px]	$\sigma_0(w(s))$ [px]	$l_{\text{mean}}^{\text{ALG\|ML}}$	$l_{\text{max}}^{\text{ALG\|ML}}$	$l_{\text{mean}}^{\text{1\|s}}$	$l_{\text{max}}^{\text{1\|s}}$
mean	27	0.31	4.83	17	0.366	0.417	1.052	1.246	1.702	2.591
max	60	0.51	14.79	77	0.851	0.980	1.169	1.918	4.118	7.477

Table 2. Data set B, cases 1 – 30. Comparing the accuracy of different homography estimates. Columns: (1) Number I of point pairs, (2–4) minimal and maximal standard deviations $\min(\sigma_x)$ and $\max(\sigma_x)$ of weighted points and their ratio, (5–6) estimated variance factors for unweighted and weighted points, (7–8) mean and maximal losses when comparing the algebraic with the ML-estimate using equal weights, $l_{\text{mean}}^{\text{ALG|ML}}$ and $l_{\text{max}}^{\text{ALG|ML}}$, and (9–10) mean and maximal losses when comparing the unweighted and the weighted ML-estimate, $l_{\text{mean}}^{\text{1|s}}$ and $l_{\text{max}}^{\text{1|s}}$

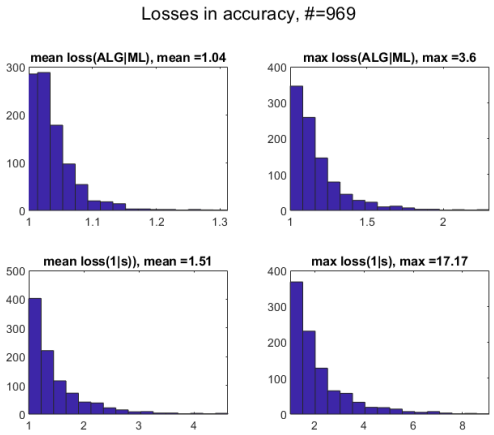

Figure 8. Data set B with 969 cases. Histograms of mean and maximum losses due to using an approximate/algebraic estimation method, and due to using equal weighting instead of scale dependent weighting

being the quadratic mean of the ϵ_i in (40).

We compared the four homographies

1. the reference (reference),
2. estimated by the algebraic minimization (alg),
3. estimated by the unweighted ML-estimation (ML 1), and
4. estimated by the scale weighted ML-estimation (ML s).

The results are shown in the *left* column of Fig. 9. Obviously all estimates lead to smaller residuals. As can be seen, the unweighted ML-estimation leads to smaller residuals, than the weighted ML-estimate (row 3: 0.45 [px] vs row 4: 0.56 [px]). This seems to be surprising, since one would expect the weighted ML-solution leads to better results. However, the result is consistent with theory, since the RMSE does not use any weighting, hence the unweighted ML-estimate needs to minimize the unweighted RMSE.

If we, therefore, analyze the weighted residuals, using a weighted root mean square error

$$\text{RMSE}_w = \sqrt{\frac{\sum_{i=1}^{I} w_i (|\mathbf{x}'_i - \mathcal{H}(\mathbf{x}_i)|^2 + |\mathbf{x}_i - \mathcal{H}^{-1}(\mathbf{x}'_i)|^2)}{8 \sum_{i=1}^{I} w_i}}, \tag{66}$$

with

$$w_i = \frac{1}{s_i^2 + s_{i'}^2}, \tag{67}$$

we obtain the histograms in the *right* column of Fig. 9. Now, as to be expected, the weighted residuals of the weighted ML-estimate are minimal (row 4: 0.4 [px]), consistent with the theoretical expectation. Also observe, all

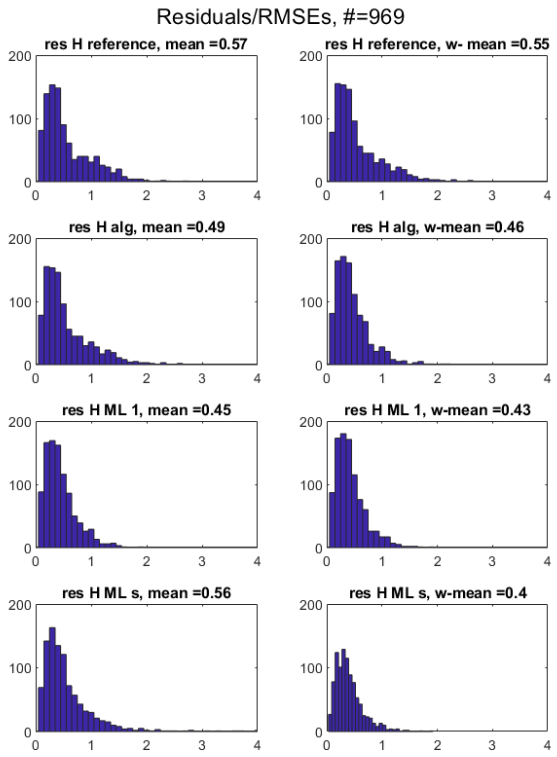

Figure 9. The residuals of the correspondences for the (**columns**) reference homography, the estimated homographies using the algebraic, the unweighted ML and the scale-weighted ML estimation. **Left column:** equally weighted residuals/RMSE, eq. (65). **Right column:** scale-weighted residuals/RMSE$_w$, eq. (66).

weighted residuals are smaller than the unweighted residuals, indicating the need to weight the coordinates used for estimating the homographies.

References

[1] Daniel Barath and Levente Hajder. A theory of point-wise homography estimation. *Pattern Recognition Letters*, 94:7–14, 2017. 3

[2] Daniel Barath, Michal Polic, Wolfgang Förstner, Torsten Sattler, Tomas Pajdla, and Zuzana Kukelova. Making affine correspondences work in camera geometry computation. In *European Conference on Computer Vision*, pages 723–740. Springer, 2020. 2

[3] Hamid Bazargani, Olexa Bilaniuk, and Robert Laganiere. A fast and robust homography scheme for real-time planar target detection. *Journal of Real-Time Image Processing*, 15(4):739–758, 2018. 2

[4] Eric Brachmann and Carsten Rother. Neural- Guided RANSAC: Learning where to sample model hypotheses. In *ICCV*, 2019. 3

[5] Luca Cavalli, Viktor Larsson, Martin Ralf Oswald, Torsten Sattler, and Marc Pollefeys. Handcrafted outlier detection revisited. In *ECCV*, 2020. 3

[6] Ondrej Chum and Jiri Matas. Matching with PROSAC-progressive sample consensus. In *Conference on Computer Vision and Pattern Recognition*, volume 1, pages 220–226. IEEE, 2005. 2

[7] Ondřej Chum and Jiří Matas. Optimal randomized RANSAC. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(8):1472–1482, 2008. 2

[8] Ondřej Chum, Jiří Matas, and Josef Kittler. Locally optimized ransac. In Bernd Michaelis and Gerald Krell, editors, *Pattern Recognition*, pages 236–243, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg. 2

[9] Ufuk Efe, Kutalmis Gokalp Ince, and Aydin Alatan. Effect of parameter optimization on classical and learning-based image matching methods. In *ICCV Workshop*, October 2021. 3

[10] Aoxiang Fan, Jiayi Ma, Xingyu Jiang, and Haibin Ling. Efficient deterministic search with robust loss functions for geometric model fitting. *IEEE TPAMI*, 2021. 2

[11] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 1

[12] Wolfgang Förstner and Kourosh Khoshelham. Efficient and Accurate Registration of Point Clouds with Plane to Plane Correspondences. In *3rd International Workshop on Recovering 6D Object Pose*, 2017. 7

[13] Wolfgang Förstner and Bernhard P. Wrobel. *Photogrammetric computer vision*. Springer, 2016. 3, 4, 6, 7

[14] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, Sept. 2020. 2

[15] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 1

[16] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 448–456, Lille, France, 07–09 Jul 2015. PMLR. 3

[17] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. *International Journal of Computer Vision*, 2020. 3

[18] Karel Lebeda, Ondřej Chum, and Jiří Matas. Fixing the locally optimized ransac. In *British Machine Vision Conference*, 2012. 2

[19] David Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 3

[20] Dmytro Mishkin, Jiri Matas, and Michal Perdoch. Mods: Fast and robust method for two-view matching. *Computer Vision and Image Understanding*, 2015. 2

[21] Rene Ranftl and Vladlen Koltun. Deep fundamental matrix estimation. In *The European Conference on Computer Vision (ECCV)*, 2018. 3

[22] E. Riba, D. Mishkin, D. Ponsa, E. Rublee, and G. Bradski. Kornia: an open source differentiable computer vision library for pytorch. In *Winter Conference on Applications of Computer Vision*, 2020. 2

[23] Peter J Rousseeuw. Least median of squares regression. *Journal of the American statistical association*, 79(388):871–880, 1984. 1

[24] Weiwei Sun, Wei Jiang, Andrea Tagliasacchi, Eduard Trulls, and Kwang Moo Yi. Attentive context normalization for robust permutation-equivariant learning. In *CVPR*, 2020. 3

[25] Stéfan van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, Tony Yu, and the scikit-image contributors. scikit-image: image processing in Python. *PeerJ*, 2:e453, 6 2014. 1

[26] Kwang Moo Yi*, Eduard Trulls*, Yuki Ono, Vincent Lepetit, Mathieu Salzmann, and Pascal Fua. Learning to find good correspondences. In *CVPR*, 2018. 3

[27] Bernhard Zeisl, Pierre Fite Georgel, Florian Schweiger, Eckehard Steinbach, and Nassir Navab. Estimation of Location Uncertainty for Scale Invariant Feature Points. In *Proc. BMVC*, pages 57.1–57.12, 2009. doi:10.5244/C.23.57. 4

[28] Jiahui Zhang, Dawei Sun, Zixin Luo, Anbang Yao, Lei Zhou, Tianwei Shen, Yurong Chen, Long Quan, and Hongen Liao. Learning two-view correspondences and geometry using order-aware network. *ICCV*, 2019. 1, 3

[29] Chen Zhao, Yixiao Ge, Feng Zhu, Rui Zhao, Hongsheng Li, and Mathieu Salzmann. Progressive correspondence pruning by consensus learning. In *ICCV*, 2021. 3