Enforcing Consistency Constraints in Uncalibrated Multiple Homography Estimation Using Latent Variables

Wojciech Chojnacki · Zygmunt L. Szpak · Michael J. Brooks · Anton van den Hengel

Abstract An approach is presented for estimating a set of interdependent homography matrices linked together by latent variables. The approach allows enforcement of all underlying consistency constraints while accounting for the arbitrariness of the scale of each individual matrix. The input data is assumed to be in the form of a set of homography matrices individually estimated from image data with no regard to the consistency constraints, appended by a set of error covariances, each characterising the uncertainty of a corresponding homography matrix. A statistically motivated cost function is introduced for upgrading, via optimisation, the input data to a set of homography matrices satisfying the constraints. The function is invariant to a change of any of the individual scales of the input matrices. The proposed approach is applied to the particular problem of estimating a set of homography matrices induced by multiple planes in the 3D scene between two views. An optimisation algorithm for this problem is developed that operates on natural underlying latent variables, with the use of those variables ensuring that all consistency constraints are satisfied. Experimental results indicate that the algorithm outperforms previous schemes proposed for the same task and is fully comparable in accuracy with the 'gold standard' bundle adjustment technique, rendering the whole approach both of practical and theoretical interest. With a view to practical application,
it is shown that the proposed algorithm can be incorporated into the familiar random sampling and consensus technique, so that the resulting modified scheme is capable of robust fitting of fully consistent homographies to data with outliers.

W. Chojnacki · Z. L. Szpak · M. J. Brooks · A. van den Hengel
School of Computer Science, The University of Adelaide, SA 5005, Australia

W. Chojnacki
E-mail: wojciech.chojnacki@adelaide.edu.au

Z. L. Szpak
E-mail: zygmunt.szpak@adelaide.edu.au

M. J. Brooks
E-mail: michael.brooks@adelaide.edu.au

A. van den Hengel
E-mail: anton.vandenhengel@adelaide.edu.au

1 Introduction

Estimation of a single homography matrix from image measurements is an important step in 3D reconstruction, mosaicing, camera calibration, metric rectification, and other tasks [24]. For some applications, like non-rigid motion detection [25, 46] or enhanced image warping [20], a whole array of homography matrices are required. The matrices have to be intrinsically interconnected to satisfy consistency constraints representing the rigidity of the motion and the scene. Moreover, the matrices have to be collectively multi-homogeneous—rescaling any individual matrix should not affect the projective information contained in the whole matrix set. A key problem in estimating multiple homography matrices is to enforce the underlying consistency constraints while accounting for the arbitrariness of the individual scales of the matrices. The need to cope with scale indeterminacy has not been particularly emphasised previously, but that it is an important aspect of the problem will be apparent from our study.

As a rule, the consistency constraints are available only in implicit form. The conventional approach to cope with such constraints is to evolve a derivative family of explicit constraints. The new constraints are typically more relaxed than the original ones. Adhering to this methodology, Shashua and Avidan [40] have found that homography

matrices induced by four or more planes in the 3D scene between two views span a four-dimensional linear subspace. Chen and Suter [4] have derived a set of strengthened constraints for the case of three or more homographies in two views. Zelnik-Manor and Irani [46] have shown that another rank-four constraint applies to a set of so-called relative homographies generated by two planes between four or more views. These latter authors have also derived constraints for larger sets of homographies and views.

Once isolated, the explicit constraints can be put to use in a procedure whereby first individual homography matrices are estimated from image data, and next these matrices are upgraded to matrices satisfying the constraints. Following this pattern, Shashua and Avidan as well as Zelnik-Manor and Irani used low-rank approximation under the Frobenius norm to enforce the rank-four constraint. Chen and Suter enforced their set of constraints also via low-rank approximation, but then employed the Mahalanobis norm with covariances of input homographies. All these estimation procedures involve input matrices coming with specific scale factors. The underlying error measures are such that a change of scale factors may *a priori* result in a different set of estimates. Furthermore, the output matrices satisfy only the derivative constraints, so their perfect consistency is not guaranteed. Another limitation of the existing methods is that each requires a certain minimum number of input homography matrices and none can work with only two such matrices.

This paper presents an alternative approach to estimating interdependent homography matrices which ensures that *all* implicit constraints are enforced and that the final estimates are unaffected by any specific choice of individual scale factors. A statistically motivated cost function is proposed for upgrading, via optimisation, the input set of homography matrices to a set satisfying all possible constraints. The function is scale change insensitive. To achieve high estimation accuracy, it incorporates the covariances of the input matrices, and this yields, upon optimisation, a statistically sound approximated maximum likelihood fit to the data. The utility of the function is demonstrated in a specific application, namely the problem of estimating a set of homography matrices induced by multiple planes in a 3D scene between two views. A variant of the Levenberg–Marquardt algorithm for that problem is developed that is specifically tailored to a parametrisation of the homography matrices via natural underlying latent variables. The use of the parametrisation ensures that all consistency constraints are satisfied. A notable contribution of this work is the development of a procedure for determining initial values of the latent variables, so that the optimisation process seeded with these values converges to a useful local minimum. Importantly, the procedure works already for two input homography matrices, hence enabling the overall estimation scheme to work for

two input homography matrices or more. The initialisation procedure is in fact of wider interest, as it is suitable for initialising other methods that operate on the same latent variables, including the canonical bundle adjustment technique for maximum likelihood estimation. The results which are contained in the experimental section of the paper validate the approach and show that the proposed estimation method outperforms existing schemes, achieving high levels of accuracy on par with the 'gold-standard' bundle adjustment technique. While the newly introduced method, like all other aforementioned methods, is not truly robust to outliers, it can—as it turns out—be made robust via incorporation into a bigger robust fitting scheme. A particular scheme of this kind forms a final contribution of the paper, and one of practical utility. It is a modification of the well-known random sampling and consensus (RANSAC) technique specialised to facilitate robust fitting of fully consistent homographies to data with outliers. The proposed estimation method enters the modified RANSAC as a computationally efficient tool for generating fully consistent homographies.

Earlier results informing this work appeared in [15]. The present findings are part of a broader, ongoing study, one of whose recent contributions is [42].

2 Multi-projective parameter estimation and latent variables

We first formulate the problem of estimating a set of interdependent homographies as a problem in multi-projective parameter estimation [14]. A general multi-projective parameter estimation problem involves a collection $\mathbf{X}_1, \ldots, \mathbf{X}_I$ of $k \times l$ matrices envisaged as data points, and a collection $\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_I$ of $k \times l$ matrices treated as parameters. Each $\mathbf{X}_i$ is assumed to be known only up to an individual multiplicative non-zero factor. The $\mathbf{\Theta}_i$'s are subject to constraints and are meant to represent improved versions of the $\mathbf{X}_i$'s. With the $k \times Il$ matrix $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_I]$ denoting the composite datum and the $k \times Il$ matrix $\mathbf{\Theta} = [\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_I]$ denoting the composite parameter, the problem under consideration is to fit $\mathbf{\Theta}$ to $\mathbf{X}$ so that the constraints on $\mathbf{\Theta}$ are met. Exemplifying this general problem is the following specific problem of interest:

Problem 1 Fit a set of 3×3 matrices, representing planar homographies engendered by various planes in a 3D scene under common projections on two images, to a given set of 3×3 matrices.

To see how the multi-projective framework applies here, consider a pair of fixed cameras with camera matrices $\mathbf{K}_1\mathbf{R}_1[\mathbf{I}_3, -\mathbf{t}_1]$ and $\mathbf{K}_2\mathbf{R}_2[\mathbf{I}_3, -\mathbf{t}_2]$. Here, the length-3 translation vector $\mathbf{t}_n$ and the 3×3 rotation matrix $\mathbf{R}_n$ represent the Euclidean transformation between the n-th ($n = 1, 2$) camera

and the world coordinate system, $\mathbf{K}_n$ is a 3×3 upper triangular calibration matrix encoding the internal parameters of the n-th camera, and, for each $m = 1, 2, \ldots, \mathbf{I}_m$ denotes the $m \times m$ identity matrix. Suppose, moreover, that a set of I planes in a 3D scene have been selected. Given $i = 1, \ldots, I$, let the i-th plane from the collection have a unit outward normal $\mathbf{n}_i$ and be situated at a distance d_i from the origin of the world coordinate system. Then, for each $i = 1, \ldots, I$, the i-th plane gives rise to a planar homography between the first and second views described by the 3×3 matrix

$$\mathbf{H}_i = w_i \mathbf{A} + \mathbf{b}\mathbf{v}_i^\top, \tag{1}$$

where

$$\begin{aligned} \mathbf{A} &= \mathbf{K}_2 \mathbf{R}_2 \mathbf{R}_1^{-1} \mathbf{K}_1^{-1}, \quad w_i = \mathbf{n}_i^\top \mathbf{t}_1 - d_i, \\ \mathbf{b} &= \mathbf{K}_2 \mathbf{R}_2 (\mathbf{t}_1 - \mathbf{t}_2), \quad \mathbf{v}_i = \mathbf{K}_1^{-\top} \mathbf{R}_1 \mathbf{n}_i. \end{aligned} \tag{2}$$

We note that in the case of calibrated cameras when one may assume that $\mathbf{K}_1 = \mathbf{K}_2 = \mathbf{I}_3$, $\mathbf{t}_1 = \mathbf{0}$, $\mathbf{R}_1 = \mathbf{I}_3$, $\mathbf{R}_2 = \mathbf{R}$, system (2) reduces to

$$\begin{aligned} \mathbf{A} &= \mathbf{R}, \quad w_i = -d_i, \\ \mathbf{b} &= \mathbf{t}, \quad \mathbf{v}_i = \mathbf{n}_i, \end{aligned} \tag{3}$$

with $\mathbf{t} = -\mathbf{R}\mathbf{t}_2$, and equality (1) becomes the familiar *direct nRt* representation

$$\mathbf{H}_i = -d_i \mathbf{R} + \mathbf{t}\mathbf{n}_i^\top$$

(cf. [2, 33]). We stress that all of our subsequent analysis concerns the general case of possibly uncalibrated cameras, with $\mathbf{A}$, $\mathbf{b}$, w_i's and $\mathbf{v}_i$'s to be interpreted according to (2) rather than (3).

Let $\mathbf{H} = [\mathbf{H}_1, \ldots, \mathbf{H}_I]$ be the composite of all the homography matrices in question. Then, with $\mathbf{a} = \text{vec}(\mathbf{A})$, where vec denotes column-wise vectorisation [32], $\boldsymbol{\eta} = [\mathbf{a}^\top, \mathbf{b}^\top, \mathbf{v}_1^\top, \ldots, \mathbf{v}_I^\top, w_1, \ldots, w_I]^\top$, and

$$\boldsymbol{\Pi}(\boldsymbol{\eta}) = [\boldsymbol{\Pi}_1(\boldsymbol{\eta}), \ldots, \boldsymbol{\Pi}_I(\boldsymbol{\eta})], \quad \boldsymbol{\Pi}_i(\boldsymbol{\eta}) = w_i \mathbf{A} + \mathbf{b}\mathbf{v}_i^\top, \tag{4}$$

$\mathbf{H}$ can be represented as

$$\mathbf{H} = \boldsymbol{\Pi}(\boldsymbol{\eta}). \tag{5}$$

The components of $\boldsymbol{\eta}$ constitute *latent variables* that link all the matrices $\boldsymbol{\Pi}_i(\boldsymbol{\eta})$ together and provide a natural parametrisation of the set of all $\mathbf{H}$'s. Since $\boldsymbol{\eta}$ has a total of $4I + 12$ entries, the set of all matrices of the form $\boldsymbol{\Pi}(\boldsymbol{\eta})$ has dimension no greater than $4I + 12$. A more refined argument shows that the set of all $\boldsymbol{\Pi}(\boldsymbol{\eta})$'s has in fact dimension equal to $4I + 7$ [12, 13]. Since $4I + 7 < 9I$ whenever $I \geq 2$, it follows that $\mathbf{H}$ resides in a proper subset of all $3 \times 3I$ matrices for $I \geq 2$. Thus, the requirement that $\mathbf{H}$ take the form as per (5) whenever $I \geq 2$ can be seen as an implicit constraint on $\mathbf{H}$, with the consequence that the $\mathbf{H}_i$'s are all interdependent. Suppose that an estimate $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_I]$ of $\mathbf{H}$ has
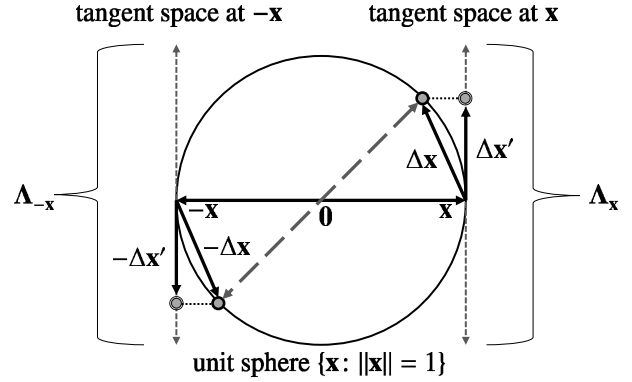

Fig. 1: Resolving scale and sign indeterminacy in the derivation of homography covariances by imposing the normalisation constraint $\|\mathbf{x}\| = 1$. The antipodal points $\mathbf{x}$ and $-\mathbf{x}$ encode one and the same homography. The true perturbations $\Delta\mathbf{x}$ and $-\Delta\mathbf{x}$ of the homography are approximated to the first order by their respective images $\Delta\mathbf{x}'$ and $-\Delta\mathbf{x}'$ via the orthogonal projections onto the tangent space to the unit sphere at $\mathbf{x}$ and $-\mathbf{x}$. Information about the spread of varying $\Delta\mathbf{x}$ is carried by the restriction of the covariance matrix $\Lambda_{\mathbf{x}}$ to the tangent space at $\mathbf{x}$. The restriction of $\Lambda_{\mathbf{x}}$ to the orthogonal complement of the tangent space at $\mathbf{x}$, which is the space spanned by $\mathbf{x}$, vanishes and carries no information about the spread of the $\Delta\mathbf{x}$'s. Likewise for $\Lambda_{-\mathbf{x}}$. Since the spread of the $\Delta\mathbf{x}$'s is the same as the spread of the $-\Delta\mathbf{x}$'s, it follows that $\Lambda_{\mathbf{x}} = \Lambda_{-\mathbf{x}}$.

been generated in some way. For example, for each i, $\mathbf{X}_i$ might be an estimate of $\mathbf{H}_i$ individually obtained from image data. The estimation problem at hand is to upgrade $\mathbf{X}$ to $\boldsymbol{\Theta} = [\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_I]$ so that $\boldsymbol{\Theta} = \boldsymbol{\Pi}(\boldsymbol{\eta})$ holds for some $\boldsymbol{\eta}$ and $\boldsymbol{\Theta}$ is close to $\mathbf{X}$ in a meaningful sense. The essence here is to find a criterion and effective means for selecting an appropriate $\boldsymbol{\eta}$.

3 Approximate maximum likelihood cost function and scale invariance

The general problem of fitting $\boldsymbol{\Theta}$ to $\mathbf{X}$ with constraints imposed on $\boldsymbol{\Theta}$ is best considered as an optimisation problem. Since the input matrices are known only up to individual scales, the output matrices should also be determined only to within individual scales. This can be achieved through the use of *multi-homogeneous* cost functions. A function J is multi-homogeneous if

$$J(\times\lambda\boldsymbol{\Theta}) = J(\boldsymbol{\Theta})$$

for each length-I vector $\lambda = [\lambda_1, \ldots, \lambda_I]^\top$ with non-zero entries, where $\times\lambda\boldsymbol{\Theta} = [\lambda_1\boldsymbol{\Theta}_1, \ldots, \lambda_I\boldsymbol{\Theta}_I]$. It is clear that if

a function is multi-homogeneous, then it is minimised not only at a single $\boldsymbol{\Theta}$ but also at all composite multiples $\times\lambda\boldsymbol{\Theta}$.

To describe a multi-homogeneous cost function relevant to our problem, for each $i = 1, \ldots, I$, let

$$\boldsymbol{\theta}_i = \mathrm{vec}(\boldsymbol{\Theta}_i), \quad \mathbf{x}_i = \mathrm{vec}(\mathbf{X}_i),$$

with each vector having length kl. Referring to the $\mathbf{X}_i$'s via their vectorisations, suppose that associated with each $\mathbf{x}_i$ is a $kl \times kl$ covariance matrix $\boldsymbol{\Lambda}_{\mathbf{x}_i}$. Our cost function will incorporate the $\boldsymbol{\Lambda}_{\mathbf{x}_i}$'s, assuming that they all take a specific form that we elaborate on next.

If the parameters of a model are represented by a vector to within a scale factor, then the covariance matrix of any particular estimate of the parameter vector is not uniquely determined. To see why, recall first that covariances are averages of squared perturbations of specific instantiations of a model. If the model is over-parametrised, involving some redundant parameters like an indeterminate scale, then the parameter vector is not identified by the model and perturbations in the model space do not translate unequivocally into perturbations in the parameter space. A way of dealing with this problem is to restrict the parameter space—that is, to identify the model—by imposing equality constraints. Such an imposition is known as *gauge fixing*, with the term "gauge" referring to any particular set of constraints. The covariances evolved with the aid of gauge-fixing rules are gauge dependent—they can look very different for different gauges [27,43]. In what follows, we eliminate the scale indeterminacy by imposing the normalisation constraint $\|\mathbf{x}\| = 1$ (see Fig. 1). Under this condition, the covariance matrix of an estimate $\mathbf{x}$ is such that $\boldsymbol{\Lambda}_{\mathbf{x}} = \boldsymbol{\Lambda}_{\pm\|\mathbf{x}\|^{-1}\mathbf{x}}$ and, moreover, $\boldsymbol{\Lambda}_{\mathbf{x}} = \mathbf{P}_{\mathbf{x}}^{\perp}\boldsymbol{\Lambda}_{\mathbf{x}}^{0}\mathbf{P}_{\mathbf{x}}^{\perp}$, where $\mathbf{P}_{\mathbf{x}}^{\perp}$ is the $kl \times kl$ symmetric projection matrix given by $\mathbf{P}_{\mathbf{x}}^{\perp} = \mathbf{I}_{kl} - \|\mathbf{x}\|^{-2}\mathbf{x}\mathbf{x}^{\top}$ and $\boldsymbol{\Lambda}_{\mathbf{x}}^{0}$ is a $kl \times kl$ symmetric matrix that we shall refer to as a *pre-covariance matrix*. An argument leading to the above assertion, together with explicit expressions for $\boldsymbol{\Lambda}_{\mathbf{x}}^{0}$ in two specific cases, can be found in Appendices A and B; see also [3, 28]. As $\mathbf{P}_{\mathbf{x}}^{\perp}\mathbf{x} = \mathbf{0}$ and $\mathbf{x}^{\top}\mathbf{P}_{\mathbf{x}}^{\perp} = \mathbf{0}^{\top}$, the matrix $\boldsymbol{\Lambda}_{\mathbf{x}}$ satisfies $\boldsymbol{\Lambda}_{\mathbf{x}}\mathbf{x} = \mathbf{0}$ and $\mathbf{x}^{\top}\boldsymbol{\Lambda}_{\mathbf{x}} = \mathbf{0}^{\top}$, and, in particular, is singular. In line with this, the companion *information matrix* associated with $\mathbf{x}$, given by the Moore–Penrose pseudo-inverse $\boldsymbol{\Lambda}_{\mathbf{x}}^{+}$ of $\boldsymbol{\Lambda}_{\mathbf{x}}$, also satisfies $\boldsymbol{\Lambda}_{\mathbf{x}}^{+}\mathbf{x} = \mathbf{0}$ and $\mathbf{x}^{\top}\boldsymbol{\Lambda}_{\mathbf{x}}^{+} = \mathbf{0}^{\top}$, and is singular.

We take for our *approximate maximum likelihood* (AML) cost function the squared *Mahalanobis distance* between any aggregate $\{\epsilon_i\|\boldsymbol{\theta}_i\|^{-1}\boldsymbol{\theta}_i\}_{i=1}^{I}$, $\epsilon_i = \pm 1$, of normalised, arbitrarily signed variants of the $\boldsymbol{\theta}_i$'s and any aggregate $\{\epsilon_i'\|\mathbf{x}_i\|^{-1}\mathbf{x}_i\}_{i=1}^{I}$, $\epsilon_i' = \pm 1$, of similar variants of the $\mathbf{x}_i$'s, with

the matrices $\boldsymbol{\Lambda}_{\mathbf{x}_i}^{+}$ serving as weights

$$J_{\mathrm{AML}}(\boldsymbol{\Theta}) = \sum_{i=1}^{I} (\epsilon_i\|\boldsymbol{\theta}_i\|^{-1}\boldsymbol{\theta}_i - \epsilon_i'\|\mathbf{x}_i\|^{-1}\mathbf{x}_i)^{\top}$$
$$\times \boldsymbol{\Lambda}_{\epsilon_i'\|\mathbf{x}_i\|^{-1}\mathbf{x}_i}^{+}(\epsilon_i\|\boldsymbol{\theta}_i\|^{-1}\boldsymbol{\theta}_i - \epsilon_i'\|\mathbf{x}_i\|^{-1}\mathbf{x}_i).$$

On account of $\boldsymbol{\Lambda}_{\mathbf{x}_i}^{+}\mathbf{x}_i = \mathbf{0}$, $\mathbf{x}_i^{\top}\boldsymbol{\Lambda}_{\mathbf{x}_i}^{+} = \mathbf{0}^{\top}$, and $\boldsymbol{\Lambda}_{\mathbf{x}_i} = \boldsymbol{\Lambda}_{\pm\|\mathbf{x}_i\|^{-1}\mathbf{x}_i}$, the expression for the function does not depend on any particular choice of the signs ϵ_i and ϵ_i' and reduces to

$$J_{\mathrm{AML}}(\boldsymbol{\Theta}) = \sum_{i=1}^{I} \|\boldsymbol{\theta}_i\|^{-2}\boldsymbol{\theta}_i^{\top}\boldsymbol{\Lambda}_{\mathbf{x}_i}^{+}\boldsymbol{\theta}_i.$$

Note that, for each $i = 1, \ldots, I$, multiplying $\boldsymbol{\theta}_i$ by a non-zero scalar λ_i results in $\boldsymbol{\theta}_i^{\top}\boldsymbol{\Lambda}_{\mathbf{x}_i}^{+}\boldsymbol{\theta}_i$ being multiplied by λ_i^2 and $\|\boldsymbol{\theta}_i\|^{-2}$ being multiplied by λ_i^{-2}, so that $\|\boldsymbol{\theta}_i\|^{-2}\boldsymbol{\theta}_i^{\top}\boldsymbol{\Lambda}_{\mathbf{x}_i}^{+}\boldsymbol{\theta}_i$ remains intact. As a result, the AML function is multi-homogeneous.

The label "approximate" in the name of J_{AML} refers to the fact that J_{AML} is an approximation—to within an additive constant—of another maximum likelihood-based cost function, namely the *exact maximum likelihood* (EML) cost function that underpins the bundle adjustment method for estimating the $\boldsymbol{\theta}_i$'s directly from image data. The EML cost function, encompassing the so-called *reprojection error* [24, Sect. 4.2.3], has for its composite argument the *principal parameters* $\boldsymbol{\theta}_i$, $i = 1, \ldots, I$, alongside some additional *nuisance parameters*. The approximation leading from the EML cost function to the AML cost function involves, among other things, the elimination of the nuisance parameters (cf. [8, 10, 11, 26, 31, 35]).

With the significance of the AML cost function elucidated, when one now takes into consideration the constraints on $\boldsymbol{\Theta}$, the corresponding constrained minimiser of J_{AML} can be viewed as a statistically well-founded estimate of $\boldsymbol{\Theta}$.

4 Rank-four constraint enforcement

This section will depart from the primary topic of the paper, which is the upgrading of multiple homographies using the AML cost function, to shed light on an earlier multi-homography updating technique commonly known as rank-four constraint enforcement. The material here is fairly technical and can be skipped without detriment to the understanding of the rest of the paper. The purpose of the departure is to reconcile an apparent contradiction: while we stress the importance of explicitly modelling the estimation process within a multi-projective framework, rank-four constraint enforcement—which is known to work—appears at first glance to be outside of this framework. The principle underpinning rank-four constraint enforcement is that a collection of five or more interrelated homography matrices resides in an at most four-dimensional subspace. In

its standard form, the technique enforces the rank-four constraint linearly via a singular value decomposition (SVD) based projection on a linear space of lower dimension. The standard form can naturally be extended to a whole family of weighted variants. It is not immediately clear how all of these, including the standard form, can fit into the category of multi-projective techniques. Settling this question seems necessary, given the multi-projective standpoint advocated in this paper. Below, we reveal that all versions of the method, despite their linear allure, are in fact fully compatible with the multi-projectiveness paradigm.

Given a composite of homography matrices $\mathbf{H} = [\mathbf{H}_1, \ldots, \mathbf{H}_I]$ satisfying (5), let H be the $9 \times I$ matrix given by

$$\mathsf{H} = [\mathbf{h}_1, \ldots \mathbf{h}_I], \quad \mathbf{h}_i = \mathrm{vec}(\mathbf{H}_i). \tag{6}$$

Since, for each $i = 1, \ldots, I$,

$$\mathbf{h}_i = w_i \, \mathrm{vec}(\mathbf{A}) + \mathrm{vec}(\mathbf{b}\mathbf{v}_i^\top) = w_i \mathbf{a} + (\mathbf{I}_3 \otimes \mathbf{b})\mathbf{v}_i, \tag{7}$$

where $\otimes$ denotes Kronecker product [32], it follows that

$$\mathsf{H} = \mathbf{S}\mathbf{T}, \tag{8}$$

where $\mathbf{S}$ is the 9×4 matrix given by

$$\mathbf{S} = [\mathbf{I}_3 \otimes \mathbf{b}, \mathbf{a}]$$

and $\mathbf{T}$ is the $4 \times I$ matrix given by

$$\mathbf{T} = \begin{bmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_I \\ w_1 & \cdots & w_I \end{bmatrix}.$$

An immediate consequence of (8) is that H has rank at most four. The requirement that H should have rank no greater than four places a genuine constraint on H, and hence also on $\mathbf{H}$, whenever $I \geq 5$. This constraint is exactly the rank-four constraint of Shashua and Avidan mentioned in the Introduction. In accordance with what has already been pointed out, when $I \geq 5$, the rank-four constraint can be enforced linearly, by employing SVD, in an infinite numbers of ways. Specifically, with every length-I vector $\mathbf{p} = [p_1, \ldots, p_I]^\top$ with non-zero entries, each having the meaning of an importance weight, there is associated a specific enforcement procedure. It needs to be immediately added that the procedures associated with different $\mathbf{p}$ are not necessarily different. If $\mathbf{p}$ and $\mathbf{p}'$ are proportional up to individual signs of the entries of $\mathbf{p}$ and $\mathbf{p}'$, that is, if $p_i = \lambda \epsilon_i p_i'$ with $\epsilon_i = \pm 1$ for some $\lambda \neq 0$ and all $i = 1, \ldots, I$, then the corresponding procedures are equivalent in the sense that the sets of respective output homographies (but not necessarily the sets of output homography matrices) are equal. If $\mathbf{p}$ and $\mathbf{p}'$ are not proportional up to individual signs of the entries of $\mathbf{p}$ and $\mathbf{p}'$, then the respective procedures are, as a rule, different.

Before describing the enforcement procedures in detail, we shall need to identify the inverse mapping r to the mapping $\mathbf{H} \mapsto \mathsf{H}$ defined in (6). It is readily verified that $\mathbf{H}$ can be expressed in terms of H as

$$\mathbf{H} = r(\mathsf{H}) = (\mathrm{vec}(\mathsf{H}))^{(3)}$$

Here, given an $n \times m$ matrix $\mathbf{S}$ and a positive integer r that divides m, $\mathbf{S}^{(r)}$ denotes the r-wise *vector transposition* of $\mathbf{S}$, that is, the $n \times (m/r)$ matrix obtained by performing a block transposition on $\mathbf{S}$, with blocks comprising length-r column vectors [19].[1] In MATLAB parlance,

$$\mathbf{H} = r(\mathsf{H}) = \texttt{reshape}(\mathsf{H}, 3, 3I).$$

With these preparations in place, given $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_I]$ with $I \geq 5$, let $\mathcal{F}_\mathbf{p}(\mathbf{X})$ be the $9 \times I$ matrix given by

$$\mathcal{F}_\mathbf{p}(\mathbf{X}) = [p_1 \|\mathbf{x}_1\|^{-1} \mathbf{x}_1, \ldots, p_I \|\mathbf{x}_I\|^{-1} \mathbf{x}_I].$$

Let $\mathcal{F}_\mathbf{p}(\mathbf{X}) = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ be the SVD of $\mathbf{X}$, with $\mathbf{D}$ being a $9 \times I$ diagonal matrix with main diagonal entries $d_{11}, \ldots d_{qq}$, $q = \min(9, I)$, and, correspondingly, let $\mathcal{F}_\mathbf{p}(\mathbf{X})_4 = \mathbf{U}\mathbf{D}_4\mathbf{V}^\top$ be the 4-truncated SVD of $\mathcal{F}_\mathbf{p}(\mathbf{X})$, with $\mathbf{D}_4$ resulting from $\mathbf{D}$ by replacing the entries $d_{55}, \ldots, d_{qq}$ by zero. Define the rank-four correction $\widehat{\Theta}_{\mathrm{rank4},\mathbf{p}}$ of $\mathbf{X}$ by

$$\widehat{\Theta}_{\mathrm{rank4},\mathbf{p}} = [\widehat{\Theta}_{\mathrm{rank4},\mathbf{p},1}, \ldots, \widehat{\Theta}_{\mathrm{rank4},\mathbf{p},I}] = r(\mathbf{U}\mathbf{D}_4\mathbf{V}^\top).$$

The procedure whereby $\mathbf{X}$ gets upgraded to $\widehat{\Theta}_{\mathrm{rank4},\mathbf{p}}$ constitutes the SVD-based method for enforcing the rank-four constraint associated with $\mathbf{p}$. The default version of the method corresponds to $\mathbf{p}$ having all entries equal to 1.

We claim that, for any $\mathbf{p}$, the mapping that sends $\mathbf{X}$ to $\widehat{\Theta}_{\mathrm{rank4},\mathbf{p}}$ has the following property: if $\mathbf{X}$ is replaced by $\times\lambda\mathbf{X}$ with λ being a length-I vector with non-zero entries, then $\widehat{\Theta}_{\mathrm{rank4},\mathbf{p}}$ is replaced by $\times\sigma\widehat{\Theta}_{\mathrm{rank4},\mathbf{p}}$ for some length-I vector σ that depends only on λ. This property implies that the homographies described by the $\widehat{\Theta}_{\mathrm{rank4},\mathbf{p},i}$'s are well defined as functions of the homographies described by the $\mathbf{X}_i$'s, providing thereby a desired reconciliation with the multi-projectiveness paradigm.

It is clear that $\mathcal{F}_\mathbf{p}(\mathbf{X})$ is *multi-positively invariant*: if $\lambda = [\lambda_1, \ldots, \lambda_I]^\top$ has all entries positive, then $\mathcal{F}_\mathbf{p}(\times\lambda\mathbf{X}) = \mathcal{F}_\mathbf{p}(\mathbf{X})$. Consequently, $\widehat{\Theta}_{\mathrm{rank4},\mathbf{p}}$ is also multi-positively invariant. We shall next show that $\widehat{\Theta}_{\mathrm{rank4},\mathbf{p}}$ is *multi-sign equivariant*: if $\epsilon = [\epsilon_1, \ldots, \epsilon_I]^\top$ is such that $\epsilon_i = \pm 1$ for each $i = 1, \ldots, I$, then

$$\widehat{\Theta}_{\mathrm{rank4},\mathbf{p}}(\times\epsilon\mathbf{X}) = \times\epsilon\widehat{\Theta}_{\mathrm{rank4},\mathbf{p}}(\mathbf{X}). \tag{9}$$

---

[1] The following examples illustrate the logic behind the definition of vector transposition:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \\ a_{41} & a_{42} \\ a_{51} & a_{52} \\ a_{61} & a_{62} \end{bmatrix}^{(2)} = \begin{bmatrix} a_{11} & a_{31} & a_{51} \\ a_{21} & a_{41} & a_{61} \\ a_{12} & a_{32} & a_{52} \\ a_{22} & a_{42} & a_{62} \end{bmatrix}, \quad \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \\ a_{41} & a_{42} \\ a_{51} & a_{52} \\ a_{61} & a_{62} \end{bmatrix}^{(3)} = \begin{bmatrix} a_{11} & a_{41} \\ a_{21} & a_{51} \\ a_{31} & a_{61} \\ a_{12} & a_{42} \\ a_{22} & a_{52} \\ a_{32} & a_{62} \end{bmatrix}.$$

To this end, we introduce $\mathbf{E} = \mathrm{diag}(\epsilon_1, \ldots, \epsilon_I)$. Note that $\mathbf{E}^\top = \mathbf{E}$ and $\mathbf{E}^2 = \mathbf{I}_I$, so in particular $\mathbf{E}$ is orthogonal: $\mathbf{E}\mathbf{E}^\top = \mathbf{I}_I$. Note also that

$$\mathcal{F}_\mathbf{p}(\times \boldsymbol{\epsilon} \mathbf{X}) = [p_1 \epsilon_1 \|\mathbf{x}_1\|^{-1} \mathbf{x}_1, \ldots, p_I \epsilon_I \|\mathbf{x}_I\|^{-1} \mathbf{x}_I] = \mathcal{F}_\mathbf{p}(\mathbf{X})\mathbf{E}.$$

Clearly,

$$\mathcal{F}_\mathbf{p}(\mathbf{X})\mathbf{E} = \mathbf{U}\mathbf{D}\mathbf{V}^\top \mathbf{E} = \mathbf{U}\mathbf{D}\mathbf{V}^\top \mathbf{E}^\top = \mathbf{U}\mathbf{D}(\mathbf{E}\mathbf{V})^\top.$$

As $\mathbf{V}$ and $\mathbf{E}$ are both orthogonal, $\mathbf{E}\mathbf{V}$ is orthogonal too, and so $\mathbf{U}\mathbf{D}(\mathbf{E}\mathbf{V})^\top$ is the SVD of $\mathcal{F}_\mathbf{p}(\mathbf{X})\mathbf{E}$. Consequently,

$$(\mathcal{F}_\mathbf{p}(\mathbf{X})\mathbf{E})_4 = \mathbf{U}\mathbf{D}_4(\mathbf{E}\mathbf{V})^\top.$$

Taking into account that $\mathbf{U}\mathbf{D}_4(\mathbf{E}\mathbf{V})^\top = \mathbf{U}\mathbf{D}_4\mathbf{V}^\top\mathbf{E}^\top = \mathbf{U}\mathbf{D}_4\mathbf{V}^\top\mathbf{E}$, we conclude that

$$r(\mathbf{U}\mathbf{D}_4\mathbf{V}^\top\mathbf{E}) = \times \boldsymbol{\epsilon} r(\mathbf{U}\mathbf{D}_4\mathbf{V}^\top),$$

which establishes (9). Being multi-positively invariant and multi-sign equivariant, the mapping $\mathbf{X} \mapsto \widehat{\boldsymbol{\Theta}}_{\mathrm{rank4},\mathbf{p}}$ has the desired property—indeed, a moment's reflection shows that $\widehat{\boldsymbol{\Theta}}_{\mathrm{rank4},\mathbf{p}}(\times \boldsymbol{\lambda} \mathbf{X}) = \times \boldsymbol{\sigma} \widehat{\boldsymbol{\Theta}}_{\mathrm{rank4},\mathbf{p}}(\mathbf{X})$, where $\boldsymbol{\sigma} = [\mathrm{sgn}(\lambda_1), \ldots, \mathrm{sgn}(\lambda_I)]^\top$. The claim is established.

We remark that the argument used above can be readily employed to verify that if $\mathbf{p}$ and $\mathbf{p}'$ are such that $p_i' = \lambda \epsilon_i p_i$, where $\lambda \neq 0$ and $\epsilon_i = \pm 1$ for each $i = 1, \ldots, I$, then the procedures $\mathbf{X} \mapsto \widehat{\boldsymbol{\Theta}}_{\mathrm{rank4},\mathbf{p}}$ and $\mathbf{X} \mapsto \widehat{\boldsymbol{\Theta}}_{\mathrm{rank4},\mathbf{p}'}$ are equivalent. Indeed, it is immediate that $\mathcal{F}_{\mathbf{p}'}(\mathbf{X}) = \mathcal{F}_\mathbf{p}(\times \boldsymbol{\xi} \mathbf{X})$, where $\boldsymbol{\xi} = [\mathrm{sgn}(\lambda)\epsilon_1, \ldots, \mathrm{sgn}(\lambda)\epsilon_I]^\top$. Hence $\widehat{\boldsymbol{\Theta}}_{\mathrm{rank4},\mathbf{p}'}(\mathbf{X}) = \times \boldsymbol{\xi} \widehat{\boldsymbol{\Theta}}_{\mathrm{rank4},\mathbf{p}}(\mathbf{X})$, and this in turn implies that the sets of homographies defined by $\widehat{\boldsymbol{\Theta}}_{\mathrm{rank4},\mathbf{p}}(\mathbf{X})$ and $\widehat{\boldsymbol{\Theta}}_{\mathrm{rank4},\mathbf{p}'}(\mathbf{X})$ coincide. One consequence of the assertion just established is that to parametrise the enforcement procedures $\mathbf{X} \mapsto \widehat{\boldsymbol{\Theta}}_{\mathrm{rank4},\mathbf{p}}(\mathbf{X})$ it suffices to consider weighting vectors $\mathbf{p}$ with all positive entries summing up to 1.

We finally point out that it is a separate question as to what $\mathbf{p}$ should be chosen to get a statistically most accurate enforcement procedure. A particular value of $\mathbf{p}$, depending on $\mathbf{X}$, can be inferred from considerations contained in [46] (reflecting the fact that interrelated homography matrices $\mathbf{X}_i$ can always be brought to the form $\mathbf{X}_i = \lambda_i(\mathbf{A} + \mathbf{b}\mathbf{v}_i^\top)$; see below), but it is conceivable that alternative choices leading to better results exist.

5 Cost function optimisation

After a detour into rank-four constraint enforcement, we now turn our attention to the question of optimisation of functions for which the AML cost function is a prototype—all this, of course, with a view to optimising the AML cost function itself.

Let J be a cost function for fitting $\boldsymbol{\Theta}$ to $\mathbf{X}$ of the form

$$J(\boldsymbol{\Theta}) = \sum_{i=1}^I \|\theta_i\|^{-2} \theta_i^\top \mathbf{A}_i \theta_i,$$

where, for each $i = 1, \ldots, I$, $\mathbf{A}_i$ is a $kl \times kl$ non-negative definite matrix. Clearly, the AML cost function conforms to this profile. Suppose that the constraints on $\boldsymbol{\Theta}$ take the form

$$\boldsymbol{\Theta} = \boldsymbol{\Pi}(\boldsymbol{\eta}), \quad \boldsymbol{\Pi}(\boldsymbol{\eta}) = [\boldsymbol{\Pi}_1(\boldsymbol{\eta}), \ldots, \boldsymbol{\Pi}_I(\boldsymbol{\eta})],$$

where $\boldsymbol{\eta}$ is a length-d vector (we have $d = 4I + 12$ in the case of the constraints given in (4)). Upon introducing the function

$$J'(\boldsymbol{\eta}) = J(\boldsymbol{\Pi}(\boldsymbol{\eta})),$$

the constrained optimisation problem in question reduces to that of optimising J', which is an *unconstrained* optimisation problem.

One way of optimising J' is to use the Levenberg–Marquardt (LM) method. The starting point is to re-express J' as

$$J'(\boldsymbol{\eta}) = \sum_{i=1}^I \|\mathbf{f}_i'(\boldsymbol{\eta})\|^2,$$

where, for each $i = 1, \ldots, I$,

$$\mathbf{f}_i'(\boldsymbol{\eta}) = \mathbf{f}_i(\pi_i(\boldsymbol{\eta})),$$
$$\mathbf{f}_i(\theta_i) = \|\theta_i\|^{-1} \mathbf{B}_i \theta_i, \quad \pi_i(\boldsymbol{\eta}) = \mathrm{vec}(\boldsymbol{\Pi}_i(\boldsymbol{\eta})),$$

with $\mathbf{B}_i$ a $kl \times kl$ matrix such that $\mathbf{B}_i^\top \mathbf{B}_i = \mathbf{A}_i$; in particular, $\mathbf{B}_i$ may be taken equal to the unique non-negative definite square root of $\mathbf{A}_i$.[2] Let $\mathbf{f}'(\boldsymbol{\eta}) = [\mathbf{f}_1'^\top(\boldsymbol{\eta}), \ldots, \mathbf{f}_I'^\top(\boldsymbol{\eta})]^\top$. The LM technique makes use of the $Ikl \times d$ Jacobian matrix $\partial_{\boldsymbol{\eta}} \mathbf{f}'$ represented as $\partial_{\boldsymbol{\eta}} \mathbf{f}' = [\partial_{\boldsymbol{\eta}} \mathbf{f}_1'^\top, \ldots, \partial_{\boldsymbol{\eta}} \mathbf{f}_I'^\top]^\top$. For each $i = 1, \ldots, I$,

$$\partial_{\boldsymbol{\eta}} \mathbf{f}_i'(\boldsymbol{\eta}) = \partial_{\theta_i} \mathbf{f}_i(\pi_i(\boldsymbol{\eta})) \partial_{\boldsymbol{\eta}} \pi_i(\boldsymbol{\eta})$$

with $\partial_{\theta_i} \mathbf{f}_i(\theta_i) = \|\theta_i\|^{-1} \mathbf{B}_i \mathbf{P}_{\theta_i}^\perp$ and $\mathbf{P}_{\theta_i}^\perp = \mathbf{I}_{kl} - \|\theta_i\|^{-2} \theta_i \theta_i^\top$. The algorithm iteratively improves on an initial approximation $\boldsymbol{\eta}_0$ to the minimiser of J' by constructing new approximations with the aid of the update rule

$$\boldsymbol{\eta}_{n+1} = \boldsymbol{\eta}_n - [\mathbf{H}(\boldsymbol{\eta}_n) + \lambda_n \mathbf{I}_d]^{-1} [\partial_{\boldsymbol{\eta}} \mathbf{f}'(\boldsymbol{\eta}_n)]^\top \mathbf{f}'(\boldsymbol{\eta}_n),$$

where $\mathbf{H} = (\partial_{\boldsymbol{\eta}} \mathbf{f}')^\top \partial_{\boldsymbol{\eta}} \mathbf{f}'$ and λ_n is a non-negative scalar that dynamically changes from step to step. Details concerning the choice of λ_n can be found in [38].

---

[2] The non-negative definite square root $\mathbf{C}^{1/2}$ of a symmetric non-negative definite matrix $\mathbf{C}$ is defined as follows: If $\mathbf{C} = \mathbf{U}\mathbf{D}\mathbf{U}^\top$ is the eigenvalue decomposition of $\mathbf{C}$ with $\mathbf{U}$ an orthogonal matrix and $\mathbf{D}$ a diagonal matrix comprising the (non-negative) eigenvalues of $\mathbf{C}$, then $\mathbf{C}^{1/2} = \mathbf{U}\mathbf{D}^{1/2}\mathbf{U}^\top$, where $\mathbf{D}^{1/2}$ is the diagonal matrix containing the square roots of the respective entries of $\mathbf{D}$.

6 Multiple homography estimation

We now proceed to consider the LM-based estimation of multiple homography matrices. We first describe the specific details of the relevant iterative scheme and then develop a suitable initialisation procedure. This procedure will be applicable not only to the LM scheme, but to any other iterative method operating on the latent variable vector η (in particular, to a bundle adjustment technique—see Section 7.2), and as such is of interest in its own right.

6.1 LM scheme

In the setup described by (4) and (5), we have, in accordance with (7),

$$\pi_i(\eta) = \text{vec}(w_i \mathbf{A} + \mathbf{b}\mathbf{v}_i^\top) = w_i \mathbf{a} + \mathbf{v}_i \otimes \mathbf{b}$$

for each $i = 1, \ldots, I$. Taking into account that $\mathbf{v}_i \otimes \mathbf{b} = (\mathbf{I}_3 \otimes \mathbf{b})\mathbf{v}_i = (\mathbf{v}_i \otimes \mathbf{I}_3)\mathbf{b}$, one readily verifies that

$$\partial_{\mathbf{a}}\pi_i = w_i \mathbf{I}_9, \qquad \partial_{\mathbf{b}}\pi_i = \mathbf{v}_i \otimes \mathbf{I}_3,$$
$$\partial_{\mathbf{v}_i}\pi_i = \mathbf{I}_3 \otimes \mathbf{b}, \qquad \partial_{\mathbf{v}_j}\pi_j = \mathbf{0} \quad (i \neq j),$$
$$\partial_{w_i}\pi_i = \mathbf{a}, \qquad \partial_{w_j}\pi_j = \mathbf{0} \quad (i \neq j).$$

Representing, for each $i = 1, \ldots, I$, $\partial_\eta \mathbf{f}_i'$ as

$$\partial_\eta \mathbf{f}_i' = [\partial_{\mathbf{a}}\mathbf{f}_i', \partial_{\mathbf{b}}\mathbf{f}_i', \partial_{\mathbf{v}_1}\mathbf{f}_i', \ldots, \partial_{\mathbf{v}_i}\mathbf{f}_i', \partial_{w_1}\mathbf{f}_i', \ldots, \partial_{w_i}\mathbf{f}_i'],$$

one finds furthermore that

$$\partial_{\mathbf{a}}\mathbf{f}_i' = w_i\|\pi_i\|^{-1}\mathbf{B}_i\mathbf{P}_{\pi_i}^\perp,$$
$$\partial_{\mathbf{b}}\mathbf{f}_i' = \|\pi_i\|^{-1}\mathbf{B}_i\mathbf{P}_{\pi_i}^\perp(\mathbf{v}_i \otimes \mathbf{I}_3),$$
$$\partial_{\mathbf{v}_i}\mathbf{f}_i' = \|\pi_i\|^{-1}\mathbf{B}_i\mathbf{P}_{\pi_i}^\perp(\mathbf{I}_3 \otimes \mathbf{b}), \quad \partial_{\mathbf{v}_j}\mathbf{f}_i' = \mathbf{0} \quad (j \neq i),$$
$$\partial_{w_i}\mathbf{f}_i' = \|\pi_i\|^{-1}\mathbf{B}_i\mathbf{P}_{\pi_i}^\perp\mathbf{a}, \qquad \partial_{w_j}\mathbf{f}_i' = \mathbf{0} \quad (j \neq i).$$

With $\partial_\eta \mathbf{f}'$ thus determined, all that is now needed is a means for determining a suitable initial value of η.

6.2 Initialisation procedure

To devise an initialisation procedure, we first consider the following problem:

Problem 2 Given $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_I]$ satisfying

$$\mathbf{X}_i = \lambda_i \mathbf{H}_i \tag{10}$$

for each $i = 1, \ldots, I$, where λ_i is a non-zero scalar and $\mathbf{H}_i = w_i \mathbf{A} + \mathbf{b}\mathbf{v}_i^\top$, solve for $\mathbf{A}$, $\mathbf{b}$, $\mathbf{v}_i$ and w_i in terms of $\mathbf{X}$.

The essence here is to recover the structure of a set of matrices which are known *a priori* to satisfy (1), but which are identified only up to (unknown) scale factors.

Observe first that the solution of the problem cannot be unique. Indeed, if $\mathbf{H}_i = w_i \mathbf{A} + \mathbf{b}\mathbf{v}_i^\top$ for each i, then also $\mathbf{H}_i = w_i' \mathbf{A}' + \mathbf{b}'\mathbf{v}_i'^\top$ for each i, where $\mathbf{A}' = \beta \mathbf{A} + \mathbf{b}\mathbf{c}^\top$, $\mathbf{b}' = \alpha\mathbf{b}$, $\mathbf{v}_i' = \alpha^{-1}\mathbf{v}_i - \alpha^{-1}\beta^{-1}\mathbf{c}$, and $w_i' = \beta^{-1}w_i$, with α and β non-zero numbers and $\mathbf{c}$ a length-3 vector $\mathbf{c}$. Therefore, we shall contend ourselves with a single specific solution.

Note that $\mathbf{X}_i = \lambda_i(w_i \mathbf{A} + \mathbf{b}\mathbf{v}_i^\top)$ implies $\mathbf{X}_i = \lambda_i'(\mathbf{A} + \mathbf{b}\mathbf{v}_i'^\top)$ with $\lambda_i' = w_i\lambda_i$ and $\mathbf{v}_i' = w_i^{-1}\mathbf{v}_i$. Thus, without loss of generality, we may assume that $w_i = 1$ for each i. Now, system (10) becomes

$$\mathbf{X}_i = \lambda_i(\mathbf{A} + \mathbf{b}\mathbf{v}_i^\top). \tag{11}$$

Select arbitrarily i and j with $i \neq j$. Taking into account that $\mathbf{v}_i^\top(\mathbf{v}_i \times \mathbf{v}_j) = \mathbf{v}_j^\top(\mathbf{v}_i \times \mathbf{v}_j) = 0$, we see that

$$\lambda_i^{-1}\mathbf{X}_i(\mathbf{v}_i \times \mathbf{v}_j) = \mathbf{A}(\mathbf{v}_i \times \mathbf{v}_j) = \lambda_j^{-1}\mathbf{X}_j(\mathbf{v}_i \times \mathbf{v}_j)$$

and further

$$\mathbf{X}_i^{-1}\mathbf{X}_j(\mathbf{v}_i \times \mathbf{v}_j) = \lambda_i^{-1}\lambda_j(\mathbf{v}_i \times \mathbf{v}_j). \tag{12}$$

Consequently, $\lambda_i^{-1}\lambda_j$ is an eigenvalue of $\mathbf{X}_i^{-1}\mathbf{X}_j$. Note that if $\mathbf{c}$ is any length-3 vector, then (11) holds with $\mathbf{A}$ replaced by $\mathbf{A} + \mathbf{b}\mathbf{c}^\top$ and with $\mathbf{b}\mathbf{v}_i^\top$ replaced by $\mathbf{b}(\mathbf{v}_i - \mathbf{c})^\top$. Accordingly, (12) holds with $(\mathbf{v}_i - \mathbf{c}) \times (\mathbf{v}_j - \mathbf{c})$ substituted for $\mathbf{v}_i \times \mathbf{v}_j$. It is easily seen that as $\mathbf{c}$ varies, the vectors

$$(\mathbf{v}_i - \mathbf{c}) \times (\mathbf{v}_j - \mathbf{c}) = (\mathbf{c} - \mathbf{v}_j) \times (\mathbf{v}_i - \mathbf{v}_j)$$

fill out a two-dimensional linear space, namely the space of all length-3 vectors orthogonal to $\mathbf{v}_i - \mathbf{v}_j$. Thus $\lambda_i^{-1}\lambda_j$ is in fact a double eigenvalue of $\mathbf{X}_i^{-1}\mathbf{X}_j$, which, generically, is unique. Denoting this eigenvalue by μ_{ij} and observing that

$$\lambda_i^{-1}\lambda_j\mathbf{X}_i - \mathbf{X}_j = \lambda_j(\lambda_i^{-1}\mathbf{X}_i - \lambda_j^{-j}\mathbf{X}_j) = \lambda_j\mathbf{b}(\mathbf{v}_i - \mathbf{v}_j)^\top,$$

we next see that $\mathbf{b}$ can be identified, up to a scalar factor, as the unique left singular vector of $\mu_{ij}\mathbf{X}_i - \mathbf{X}_j$ corresponding to a non-zero singular value.

Having made these observations, we now fix i_0 arbitrarily. For each $i \neq i_0$, let μ_{ii_0} be the double eigenvalue of $\mathbf{X}_i^{-1}\mathbf{X}_{i_0}$. Then, for each $i \neq i_0$,

$$\mu_{ii_0}\mathbf{X}_i = \lambda_i^{-1}\lambda_{i_0}\mathbf{X}_i = \lambda_{i_0}(\mathbf{A} + \mathbf{b}\mathbf{v}_i^\top).$$

Relabelling $\lambda_{i_0}\mathbf{A}$ as $\mathbf{A}$ and $\lambda_{i_0}\mathbf{b}$ as $\mathbf{b}$, the last relation simplifies to

$$\mu_{ii_0}\mathbf{X}_i = \mathbf{A} + \mathbf{b}\mathbf{v}_i^\top \tag{13}$$

and the i_0-th defining equation becomes

$$\mathbf{X}_{i_0} = \mathbf{A} + \mathbf{b}\mathbf{v}_{i_0}^\top. \tag{14}$$

Algorithm 1: INITIALISATION-PROCEDURE

Input: $\mathbf{X}_i$, $i = 1, \ldots, I$
Output: $\bar{\mathbf{a}}$, $\bar{\mathbf{b}}$, $\bar{\mathbf{v}}_i$ and $\bar{w}_i$, $i = 1, \ldots, I$

1 For each $i = 1, \ldots, I$, let $\bar{w}_i = 1$.
2 Select i_0 arbitrarily from the range between 1 and I.
3 For each $i \neq i_0$, determine two closest eigenvalues $\mu_{ii_0}^{(1)}$ and $\mu_{ii_0}^{(2)}$
    of $\mathbf{X}_i^{-1}\mathbf{X}_{i_0}$, and set $\mu_{ii_0} = (\mu_{ii_0}^{(1)} + \mu_{ii_0}^{(2)})/2$.
4 Take for $\bar{\mathbf{b}}$ a left singular vector of the $3 \times 6(I-1)$ juxtaposition
    (horizontal concatenation) of the matrices $\mu_{ii_0}^{(1)}\mathbf{X}_i - \mathbf{X}_{i_0}$ and
    $\mu_{ii_0}^{(2)}\mathbf{X}_i - \mathbf{X}_{i_0}$, $i \neq i_0$, corresponding to the biggest singular value.
5 For each $i \neq i_0$, replace μ_{ii_0} with the real part of μ_{ii_0}. Also
    replace $\bar{\mathbf{b}}$ with the vector comprising the real parts of the
    elements of $\bar{\mathbf{b}}$.
6 Let $\bar{\mathbf{a}} = \text{vec}(\mathbf{X}_{i_0})$ and $\bar{\mathbf{v}}_{i_0} = \mathbf{0}$.
7 For each $i \neq i_0$, set $\bar{\mathbf{v}}_i = \bar{\mathbf{v}}_{i_0} + \|\bar{\mathbf{b}}\|^{-2}(\mu_{ii_0}\mathbf{X}_i - \mathbf{X}_{i_0})^\top\bar{\mathbf{b}}$.

Take for $\mathbf{b}$ the left singular vector of the $3 \times 3(I-1)$ matrix obtained by juxtapositioning (concatenating horizontally) the matrices $\mu_{ii_0}\mathbf{X}_i - \mathbf{X}_{i_0}$, $i \neq i_0$, corresponding to a single non-zero singular value. Now, by subtracting (14) from (13), we find, for each $i \neq i_0$,

$$\mu_{ii_0}\mathbf{X}_i - \mathbf{X}_{i_0} = \mathbf{b}(\mathbf{v}_i - \mathbf{v}_{i_0})^\top$$

whence

$$\mathbf{v}_i = \mathbf{v}_{i_0} + \|\mathbf{b}\|^{-2}(\mu_{ii_0}\mathbf{X}_i - \mathbf{X}_{i_0})^\top\mathbf{b}.$$

To fully solve our problem, it remains to determine $\mathbf{v}_{i_0}$ and $\mathbf{A}$. The only constraint that we now have to take into account is (14). One simple solution is

$$\mathbf{A} = \mathbf{X}_{i_0}, \quad \mathbf{v}_{i_0} = \mathbf{0}.$$

With our problem successfully solved, we are now in position to furnish a desired initialisation procedure. Based on a noisy data set $\mathbf{X}$, a seed $\bar{\boldsymbol{\eta}} = [\bar{\mathbf{a}}^\top, \bar{\mathbf{b}}^\top, \bar{\mathbf{v}}_1^\top, \ldots, \bar{\mathbf{v}}_I^\top, \bar{w}_1, \ldots, \bar{w}_I]^\top$ for any iterative method operating on $\boldsymbol{\eta}$ is obtained by modifying the specific solution given above. The modification reflects the fact that $\mathbf{X}$ admits only an approximate representation as in (10). The steps of the initialisation procedure are detailed in Algorithm 1.

It is noteworthy that for the above procedure to work, just two different input homographies suffice. This is in contrast with the initialisation proposed by Chen and Suter [4] which requires at least three different homographies.

7 Experimental verification

The method was tested on both synthetic and real data. Synthetic data were used to quantify the effect of noise on the method. Real data were used to evaluate the performance of the method in a real-world situation.
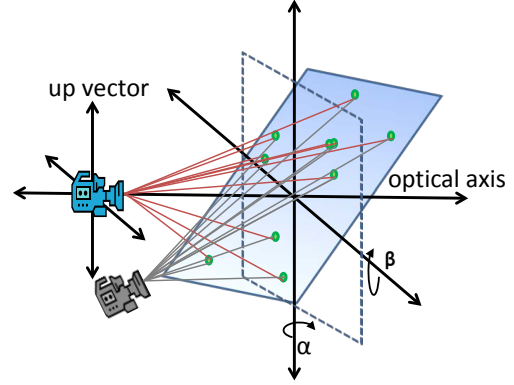

Fig. 2: Synthetic data generation procedure.

7.1 Synthetic data

Synthetic data were created by generating true corresponding points for some stereo configuration and adding random Gaussian noise to these points. Many configurations were investigated. Any specific instantiation of true image points was developed as follows. First, we chose a realistic geometric configuration for two cameras. Next, we applied a random rotation and translation to a plane that is parallel to the first camera's image plane (see Fig. 2). Repeating this last step several times, we generated several planes in the 3D scene. Finally, between 25 to 50 points in each plane were randomly selected in the field of view of both cameras, and these were projected onto two 640×480 pixel images to provide true image points.

7.2 Synthetic simulation procedure

Each synthetic true image point was perturbed by independent homogeneous Gaussian noise at a preset level. For different series of experiments, different noise levels were applied. This resulted in I groups of noise-contaminated pairs of corresponding points $\{\mathbf{m}_{n,i}, \mathbf{m}'_{n,i}\}_{n=1}^{N_i}$, $i = 1, \ldots, I$, corresponding to I different planes in the 3D scene, with $I \in \{2, 4, 5, 6, 7, 8\}$ for different experiments; here, N_i is the number of feature points in the i-th plane.

The estimation methods considered were:

- *DLT*   direct linear transform,
- *FNS*   fundamental numerical scheme,
- *WALS*   weighted alternating least squares,
- *AML*   approximate maximum likelihood,
- *BA*   bundle adjustment.

DLT [24] is a linear method for estimating a single homography and FNS [8, 39] is an iterative method for the same purpose, the two methods optimising two different, yet related cost functions. For the small noise levels utilised in

(a) 200 simulations and 4 planes.
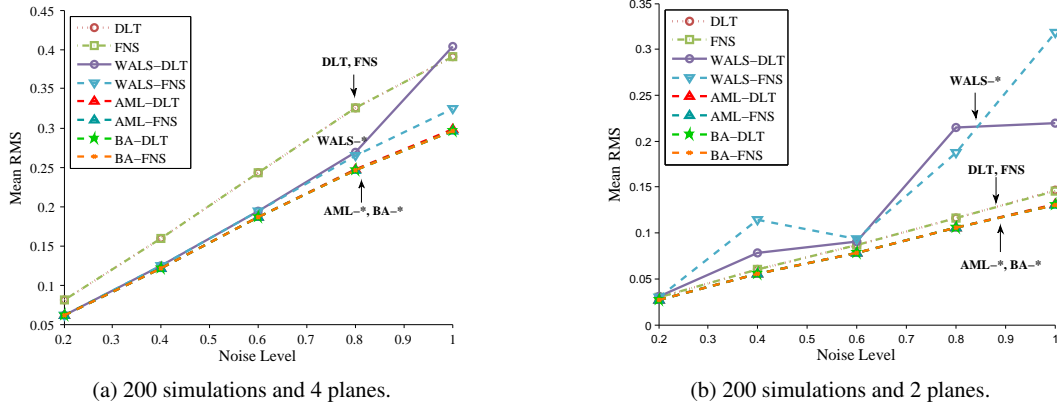
(b) 200 simulations and 2 planes.

Fig. 3: Comparison of homography estimation methods on synthetic data. The results are based on averaging the reprojection error over all planes in the scene. The DLT and FNS estimators do not enforce consistency constraints, while WALS, AML, and our variant of BA do. The results show that enforcing consistency constraints can result in considerable denoising. The typical performance of AML and BA is virtually indistinguishable. The poor performance of WALS in (b) is due to the instability of the method. On some occasions it converges to a very poor solution with high reprojection error. This adversely affects its Mean RMS score.

our study, FNS produces the same results as single homography bundle adjustment, and to all intents and purposes may be regarded as the gold standard single homography estimation method. Both DLT and FNS were run on suitably normalised data. For each $i = 1, \ldots, I$, two data-dependent normalisation matrices $\mathbf{T}_i$ and $\mathbf{T}'_i$ were applied to $\{\mathbf{m}_{n,i}, \mathbf{m}'_{n,i}\}_{n=1}^{N_i}$ using the rule

$$\tilde{\mathbf{m}}_{n,i} = \mathbf{T}_i \mathbf{m}_{n,i} \quad \text{and} \quad \tilde{\mathbf{m}}'_{n,i} = \mathbf{T}'_i \mathbf{m}'_{n,i}$$

to produce individually normalised corresponding points $\{\tilde{\mathbf{m}}_{n,i}, \tilde{\mathbf{m}}'_{n,i}\}_{n=1}^{N_i}$ à la Hartley [7, 9, 23]. These normalised groups were used as input to DLT and FNS to produce two sets of I homography matrices $\{\widehat{\tilde{\boldsymbol{\Theta}}}_{\text{DLT},i}\}_{i=1}^{I}$ and $\{\widehat{\tilde{\boldsymbol{\Theta}}}_{\text{FNS},i}\}_{i=1}^{I}$. The final estimates $\widehat{\boldsymbol{\Theta}}_{\text{DLT}} = \{\widehat{\boldsymbol{\Theta}}_{\text{DLT},i}\}_{i=1}^{I}$ and $\widehat{\boldsymbol{\Theta}}_{\text{FNS}} = \{\widehat{\boldsymbol{\Theta}}_{\text{FNS},i}\}_{i=1}^{I}$ were obtained by applying, for each $i = 1, \ldots, I$, the back-transformation $\boldsymbol{\Theta}_i \mapsto \mathbf{T}'^{-1}_i \boldsymbol{\Theta}_i \mathbf{T}_i$ to $\widehat{\tilde{\boldsymbol{\Theta}}}_{\text{DLT},i}$ and $\widehat{\tilde{\boldsymbol{\Theta}}}_{\text{FNS},i}$, respectively.

WALS is Chen and Suter's method of weighted alternating least squares with weights derived from the covariances of input estimates. Both WALS and our proposed AML method use as input, in one version, the estimates produced by FNS and, in another version, the estimates produced by DLT. For reasons of numerical stability, both WALS and AML required data normalisation as a pre-processing step. A common data normalisation procedure was employed to both methods, namely we combined all the points together to produce two global normalisation matrices $\mathbf{T}$ and $\mathbf{T}'$. These matrices were used to produce globally normalised corresponding points $\{\mathring{\mathbf{m}}_{n,i}, \mathring{\mathbf{m}}'_{n,i}\}_{n=1}^{N_i}$, $i = 1, \ldots, I$, defined by

$$\mathring{\mathbf{m}}_{n,i} = \mathbf{T}\mathbf{m}_{n,i} \quad \text{and} \quad \mathring{\mathbf{m}}'_{n,i} = \mathbf{T}'\mathbf{m}'_{n,i}.$$

The input of the FNS-initialised versions of WALS and AML, WALS-FNS and AML-FNS, took the form of the FNS estimates based on $\{\mathring{\mathbf{m}}_{n,i}, \mathring{\mathbf{m}}'_{n,i}\}_{n=1}^{N_i}$, $i = 1, \ldots, I$. Likewise, the input of the DLT-initialised versions of WALS and AML, WALS-DLT and AML-DLT, took the form of the DLT estimates based on $\{\mathring{\mathbf{m}}_{n,i}, \mathring{\mathbf{m}}'_{n,i}\}_{n=1}^{N_i}$, $i = 1, \ldots, I$. For each initialisation, a specific prescription, described below, was used to generate the pre-covariance matrix $\boldsymbol{\Lambda}^0_{\mathbf{x}_i}$ for the vectorisation $\mathbf{x}_i$ of $\mathbf{X}_i$ for each $i = 1, \ldots, I$. The matrices $\mathbf{X}_i$ and pre-covariances $\boldsymbol{\Lambda}^0_{\mathbf{x}_i}$ were next used as input to the WALS and AML methods to produce estimates which, upon applying the back-transformation $\boldsymbol{\Theta} \mapsto \mathbf{T}'^{-1}\boldsymbol{\Theta}\mathbf{T}$, were taken to be $\widehat{\boldsymbol{\Theta}}_{\text{WALS}} = \{\widehat{\boldsymbol{\Theta}}_{\text{WALS},i}\}_{i=1}^{I}$ and $\widehat{\boldsymbol{\Theta}}_{\text{AML}} = \{\widehat{\boldsymbol{\Theta}}_{\text{AML},i}\}_{i=1}^{I}$, respectively

In the case of the FNS-initialised methods, the pre-covariance matrix for the vectorisation $\mathbf{x}_i$ of $\mathbf{X}_i$, based on $\{\mathring{\mathbf{m}}_{n,i}, \mathring{\mathbf{m}}'_{n,i}\}_{n=1}^{N_i}$, was taken to be

$$\boldsymbol{\Lambda}^0_{\mathbf{x}_i} = (\mathbf{M}_{\mathbf{x}_i})^+_8, \tag{15}$$

$$\mathbf{M}_{\mathbf{x}_i} = \|\mathbf{x}_i\|^2 \sum_{n=1}^{N_i} \mathbf{U}(\mathring{\mathbf{z}}_{n,i})(\boldsymbol{\Sigma}(\mathring{\mathbf{z}}_{n,i}, \mathbf{x}_i))^+_2 \mathbf{U}(\mathring{\mathbf{z}}_{n,i})^\top. \tag{16}$$

Here, $\mathring{\mathbf{z}}_{n,i} = [\mathring{u}_{n,i}, \mathring{v}_{n,i}, \mathring{u}'_{n,i}, \mathring{v}'_{n,i}]^\top$ is the result of combining $\mathring{\mathbf{m}}_{n,i} = [\mathring{u}_{n,i}, \mathring{v}_{n,i}, 1]^\top$ and $\mathring{\mathbf{m}}'_{n,i} = [\mathring{u}'_{n,i}, \mathring{v}'_{n,i}, 1]^\top$ into a single vector; $\mathbf{U}(\mathring{\mathbf{z}}_{n,i})$ and $\boldsymbol{\Sigma}(\mathring{\mathbf{z}}_{n,i}, \mathbf{x}_i)$ are defined by

$$\mathbf{U}(\mathbf{z}) = -\mathbf{m} \otimes [\mathbf{m}']_\times,$$

$$\mathbf{B}(\mathbf{z}) = \partial_{\mathbf{z}} \text{vec}(\mathbf{U}(\mathbf{z})) \boldsymbol{\Lambda}_{\mathbf{z}} [\partial_{\mathbf{z}} \text{vec}(\mathbf{U}(\mathbf{z}))]^\top,$$

$$\boldsymbol{\Sigma}(\mathbf{z}, \mathbf{x}) = (\mathbf{I}_3 \otimes \mathbf{x}^\top)\mathbf{B}(\mathbf{z})(\mathbf{I}_3 \otimes \mathbf{x})$$

with $\mathbf{z} = [u, v, u', v']^\top$ derived from $\mathbf{m} = [u, v, 1]^\top$ and $\mathbf{m}' = [u', v', 1]^\top$, and with $\mathbf{x}$ a length-9 vector; for a length-3 vector

a, $[\mathbf{a}]_\times$ is the 3×3 anti-symmetric matrix given by

$$[\mathbf{a}]_\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix};$$

and $\mathbf{A}_r^+$ denotes the *rank-r truncated pseudo-inverse* of the matrix $\mathbf{A}$ (see [26] or Appendix A for the definition). The image data covariance matrices $\Lambda_{\mathring{\mathbf{z}}_{n,i}}$, incorporated in the matrices $\Sigma(\mathring{\mathbf{z}}_{n,i}, \mathbf{x}_i)$, were all chosen to be in their default form $\mathrm{diag}(1,1,1,1)$, corresponding to isotropic homogeneous noise in image point measurement. The details of formula (15) are given in Appendix A.

The DLT-initialised methods employed the raw covariance matrix for the vectorisation $\mathbf{x}_i$ of $\mathbf{X}_i$, based on $\{\mathring{\mathbf{m}}_{n,i}, \mathring{\mathbf{m}}'_{n,i}\}_{n=1}^{N_i}$, in the form

$$\Lambda_{\mathbf{x}_i}^0 = (\mathbf{M}_{\mathbf{x}_i})_8^+ \mathbf{D}_{\mathbf{x}_i} (\mathbf{M}_{\mathbf{x}_i})_8^+. \tag{17}$$

with $\mathbf{M}_{\mathbf{x}_i}$ given in (16) and

$$\mathbf{D}_{\mathbf{x}_i} = \|\mathbf{x}_i\|^{-2} \sum_{n=1}^{N} \mathbf{U}(\mathring{\mathbf{z}}_{n,i}) \Sigma(\mathring{\mathbf{z}}_{n,i}, \mathbf{x}_i) \mathbf{U}(\mathring{\mathbf{z}}_{n,i})^\top.$$

Formula (17) relies on the material presented in Appendix B.

BA is a method for calculating maximum likelihood estimates and is regarded as the gold standard for performing optimal parameter estimation. It is the same as what is called *joint* bundle adjustment in [42]. The qualifier "joint" serves to emphasise the fact that this variant of bundle adjustment, using the latent variables, enforces full homography consistency constraints. In contrast, naive *separate* bundle adjustment minimises the gold standard geometric error, but does not enforce consistency constraints. In this paper we do not explicitly consider separate bundle adjustment, therefore we label joint bundle adjustment simply as BA. However, separate bundle adjustment enters implicitly into our considerations through FNS, as, in accordance with our earlier remark, FNS applied to estimate each individual homography separately yields results indistinguishable from those produced by separate bundle adjustment for the levels of noise employed in our experiments.

The BA method was used in two variants, BA-FNS and BA-DLT, which differed in the way the method's iterative process was initialised. In both variants, a BA estimate $\widehat{\Theta}_{\mathrm{BA}} = \{\widehat{\Theta}_{\mathrm{BA},i}\}_{i=1}^I$ was generated directly from all the image data $\{\{\mathring{\mathbf{m}}_{n,i}, \mathring{\mathbf{m}}'_{n,i}\}_{n=1}^{N_i}\}_{i=1}^I$ by minimising the *reprojection error*

$$\sum_{i=1}^{I} \sum_{n=1}^{N_i} \left(d(\mathring{\mathbf{m}}_{n,i}, \underline{\mathbf{m}}_{n,i})^2 + d(\mathring{\mathbf{m}}'_{n,i}, \Pi_i(\eta)\underline{\mathbf{m}}_{n,i})^2 \right)$$

over all choices of parameter vectors η and 2D points $\{\{\underline{\mathbf{m}}_{n,i}\}_{n=1}^{N_i}\}_{i=1}^I$, with the minimum attained at the composite of $\widehat{\eta}$ and $\{\{\widehat{\underline{\mathbf{m}}}_{n,i}\}_{n=1}^{N_i}\}_{i=1}^I$ resulting in $\widehat{\Theta}_{\mathrm{BA},i} = \Pi_i(\widehat{\eta})$. Here, $d(\mathbf{m}, \mathbf{n})$ denotes the Euclidean distance between the points $\mathbf{m}$ and $\mathbf{n}$

dehomogenised at the last, third entry. The initial value of each $\underline{\mathbf{m}}_{n,i}$ was taken to be $\mathring{\mathbf{m}}_{n,i}$ in both variants, and the initial value of η was obtained from the result of AML-FNS for BA-FNS, and from the result of AML-DLT for BA-DLT. Upon initialisation, η and the $\underline{\mathbf{m}}_{n,i}$'s were recomputed iteratively by an LM scheme adapted to the task of minimising the error given above. With $\widehat{\mathbf{A}}_{\mathrm{BA}}$, $\widehat{\mathbf{b}}_{\mathrm{BA}}$, $\widehat{\mathbf{v}}_{\mathrm{BA},i}$'s, $\widehat{w}_{\mathrm{BA},i}$'s derived from the terminal value of η, the estimates $\widehat{\Theta}_{\mathrm{BA},i}$ were finally obtained by applying the back-transformation $\Theta \mapsto \mathbf{T}'^{-1}\Theta\mathbf{T}$ to $\widehat{w}_{\mathrm{BA},i}\widehat{\mathbf{A}}_{\mathrm{BA}} + \widehat{\mathbf{b}}_{\mathrm{BA}}\widehat{\mathbf{v}}_{\mathrm{BA},i}^\top$.

7.3 Real data

To investigate the performance of the proposed method on real data, we utilised the Model House sequence from the Oxford dataset,[3] the Lady Symon and Old Classics Wing scenes from the AdelaideRMF dataset[4] [45], and additionally two traffic scenes from the Traffic Signs dataset[5] [30].

We generated corresponding points for the Oxford and Traffic Signs datasets using correlation-based matching on Harris corner points. Because the matching of corner points was done without manual intervention, the resulting set of corresponding points included pure outliers (incorrect matches). Corresponding points for the AdelaideRMF dataset were generated in a different manner, but also included pure outliers. The curator of the AdelaideRMF dataset detected key points using SIFT and generated true correspondences by manually matching only a subset of key points. The remaining key points were randomly matched and labelled as pure outliers.

To make the experiments on real data as realistic as possible, we did not manually group data points into different planar regions. Instead, we took advantage of recent progress in robust hypothesis generation for multi-structure data [6] and sampled a thousand candidate homographies based on four-element sets of matched correspondences. For each pair of corresponding points, we computed the *symmetric transfer error* [24, Sect. 4.2.2] for all 1,000 homographies and ranked the homographies according to this error. This means that each correspondence had an associated preference list that ranked the homographies. Making use of the *ordered residual kernel* method [5] which defines a distance between rankings, together with an agglomerating clustering scheme, we grouped the pairs of corresponding points into several clusters. The clustering scheme that we applied does not guarantee that all final clusters belong to different planes in the scene. In fact, frequently correspondences are fragmented into two different clusters even
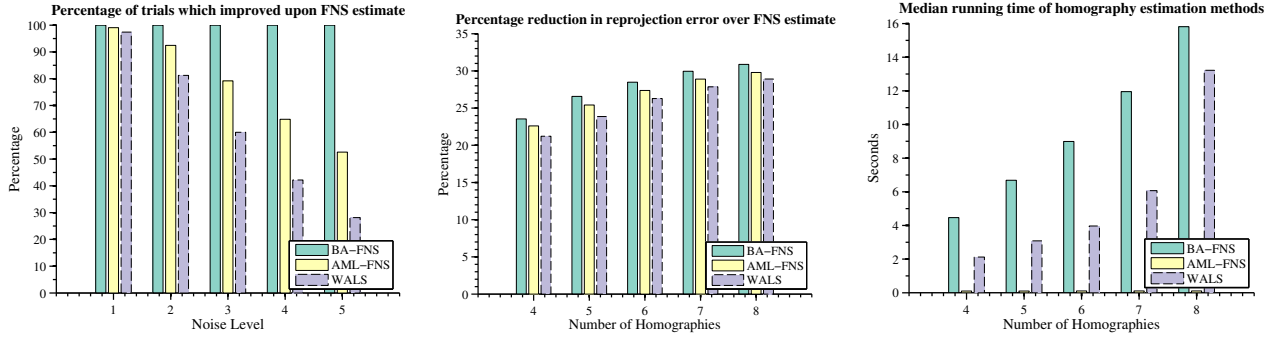
(a) Measuring the percentage of trials for which various joint homography estimation methods converged to a solution with lower reprojection error than FNS. Results are based on 1500 trials with 4 homographies per trial.

(b) Improvement over FNS for estimation methods enforcing consistency constraints, expressed in terms of percentage reduction in reprojection error. Results are based on 1500 trials, with 50 data points per trial and the noise level of $\sigma = 2$ pixels.

(c) Comparison of median running time of various homography estimation methods. Results are based on 1500 trials, with 50 data points per trial and the noise level of $\sigma = 2$ pixels. Note that the running time for AML-FNS was less than a tenth of a second, and so the AML-FNS bar graphs are barely visible.

Fig. 4: Evaluating homography estimation methods on synthetic data by: (a) comparing how often the consistency constraint enforcing methods yield lower reprojection errors than separate FNS homography estimation; (b) quantifying the expected percentage reduction in reprojection error, and (c) measuring their running times.

though visually they should belong to the same plane. This means that effectively the same planar region may have *more than one* homography associated with it. Such is the case with the Model House sequence where clustering led to the appearance of six homographies (corresponding to six clusters) even though there are only *four* actual planes in the scene. The fragmentation effect is not unique to our clustering scheme and was also observed by Fouhey et al. [18], who used a different algorithm (J-linkage) to group planar data points together. The empirical fact that fragmentation occurs so frequently serves as an affirmation that it is important to exploit consistency constraints as a means for enhancing coherency between homographies derived from logically linked clusters.

7.4 Real data experiment procedure

After automatically grouping the data points into several clusters, we estimated homographies using DLT for each group. To compare and contrast the stability and accuracy of the consistency enforcing estimation methods, we used the same initialisation procedure (Algorithm 1) on the DLT estimates of each group to seed AML-DLT, WALS-DLT, and BA-DLT.

7.5 Quantitative comparison of methods

On synthetic data, for $\widehat{\boldsymbol{\Theta}} = \{\widehat{\boldsymbol{\Theta}}_i\}_{i=1}^I$ generated by the DLT, FNS, WALS, AML, and BA methods, the common distance

used to quantify data–model discrepancies was the *mean root-mean-square (RMS) reprojection error from truth*

$$\frac{1}{I} \sum_{i=1}^{I} \sqrt{ \frac{1}{4N_i K} \sum_{k=1}^{K} \min_{\underline{\mathbf{m}}_{n,i}^{(k)}} \sum_{n=1}^{N_i} \left(d(\overline{\mathbf{m}}_{n,i}^{(k)}, \underline{\mathbf{m}}_{n,i}^{(k)})^2 + d(\overline{\mathbf{m}}_{n,i}'^{(k)}, \widehat{\boldsymbol{\Theta}}_i \underline{\mathbf{m}}_{n,i}^{(k)})^2 \right) },$$

where K is the number of experiments, and, for each $k = 1, \ldots, K$, $\{\{\overline{\mathbf{m}}_{n,i}^{(k)}, \overline{\mathbf{m}}_{n,i}'^{(k)}\}_{n=1}^{N_i}\}_{i=1}^I$ are noiseless data and $\{\{\underline{\mathbf{m}}_{n,i}^{(k)}\}_{n=1}^{N_i}\}_{i=1}^I$ are arbitrary 2D points over which the minimum is taken in the k-th experiment. A comparison of the methods on synthetic data is shown in Figs. 3 and 4. The results indicate that the proposed algorithm outperforms DLT, FNS, WALS-DLT, and WALS-FNS, and is practically indistinguishable in performance from the BA method for small noise levels. For larger noise levels all algorithms that enforce consistency constraints become susceptible to converging to a poor local minimum, resulting in homography estimates that are worse than separately estimated homographies. Figure 4a shows comparatively how often the algorithms that enforce consistency homographies managed to improve upon the (separately) FNS-estimated homographies. While the likelihood of converging to a superior solution decreased as the noise level increased, our method still fared better than WALS. With $\sigma = 2$ pixels our algorithm converged to a superior solution in more than 90% of trials. The deterioration of the algorithm at higher noise levels can be attributed to Algorithm 1 which is used to find initial values for the latent variables. For high noise levels Algorithm 1 yields latent variables that are associated with high cost function values, which means that the optimisation methods are initialised far from the optimal solution and

(a) view 0.

(b) view 1.

(c) Reprojection error for view 0 and 1.

(d) view 2.

(e) view 3.

(f) Reprojection error for view 2 and 3.

(g) view 4.

(h) view 5.
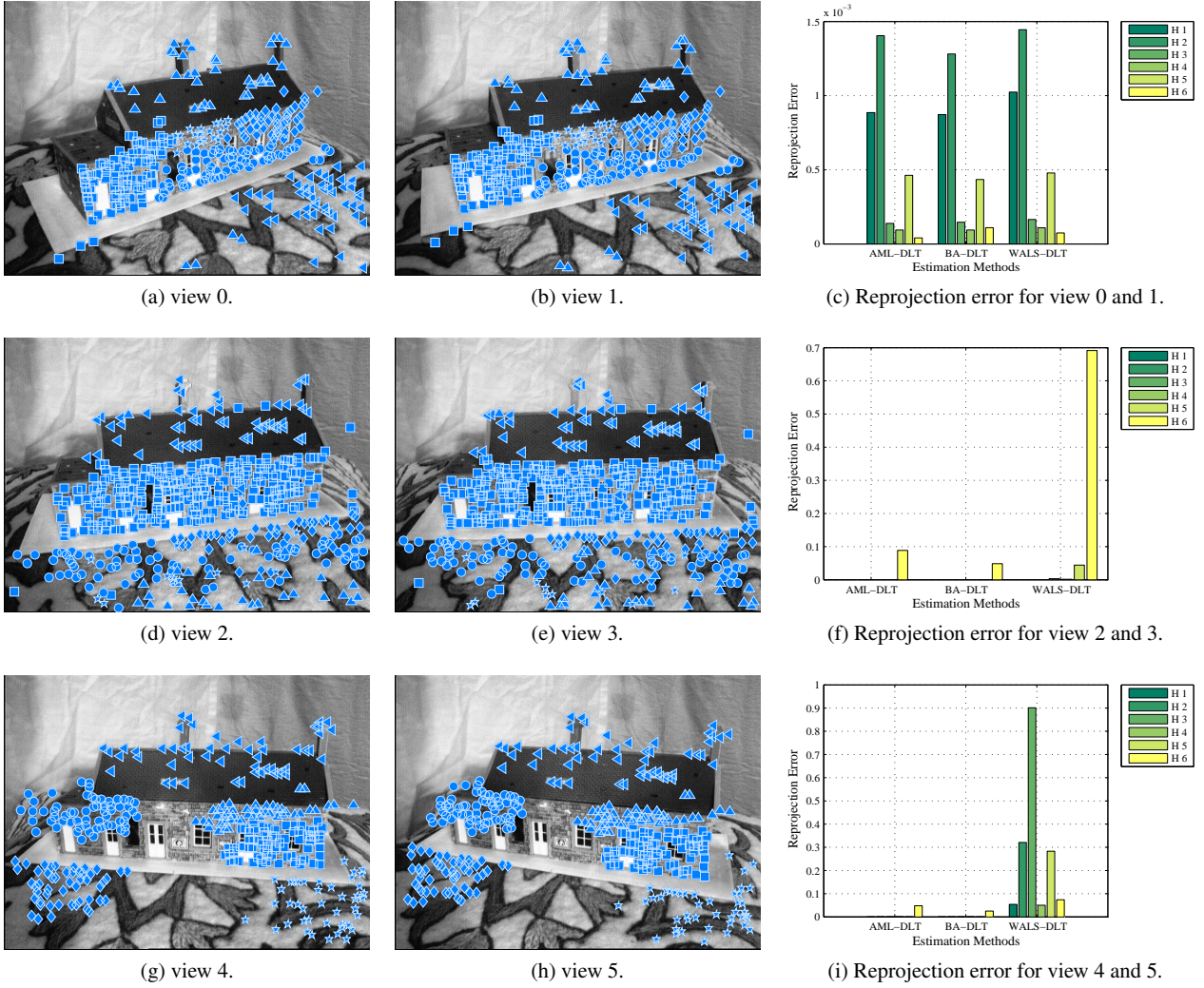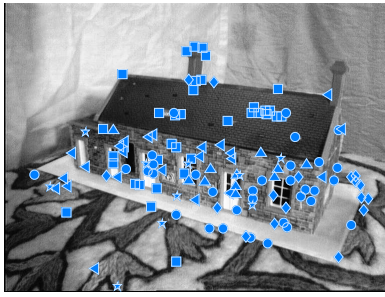

(i) Reprojection error for view 4 and 5.

Fig. 5: Comparison of consistency enforcing estimation methods on the Model House sequence. The first two columns show the feature points (inliers) associated with various homographies on the model house views. The third column compares the reprojection error of AML-DLT, WALS-DLT, and BA-DLT for each homography. Note that in some figures the errors associated with particular homographies are so close to zero that the bar graphs are barely visible.

have a greater chance of stopping at a poor local minimum. In Fig. 4b, we display the percentage reduction in reprojection error that enforcing consistency constraints can produce when compared with separate homography estimation. The results show that one can expect approximately 23% reduction for four homographies, and up to approximately 30% reduction for eight homographies. Even when there were only two planes in the scene, we still observed an approximately 10% reduction in reprojection error. These findings evidence that upgrading a set of inconsistent homographies to a fully consistent set can yield considerable practical benefits and constitute the *raison d'être* of our algorithm. In Fig. 4c, we present the median running time of BA, WALS, and our method. Our algorithm is orders of magnitude faster than all other options, requiring on average only four itera-

tions to converge. It is thanks to our algorithm's remarkable and unique computational efficiency that it can be incorporated into a random sampling and consensus framework to filter out sets of putative homographies that are mutually incompatible (see Section 7.6).

On real data, we evaluate the performance of AML-DLT, WALS-DLT, and BA-DLT against each other by comparing the quality of each estimated homography separately. The comparisons are made by plotting bar graphs of the *RMS reprojection error from data*
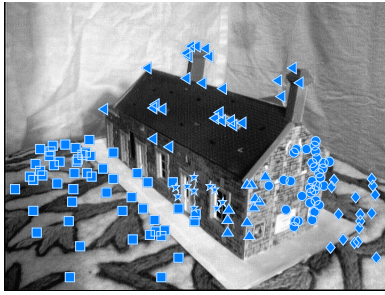
$$\sqrt{\frac{1}{4N_i} \min_{\underline{\mathbf{m}}_{n,i}} \sum_{n=1}^{N_i} \left(d(\mathbf{m}_{n,i}, \underline{\mathbf{m}}_{n,i})^2 + d(\mathbf{m}'_{n,i}, \widehat{\boldsymbol{\Theta}}_i \underline{\mathbf{m}}_{n,i})^2 \right)},$$
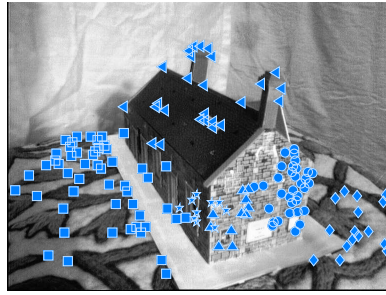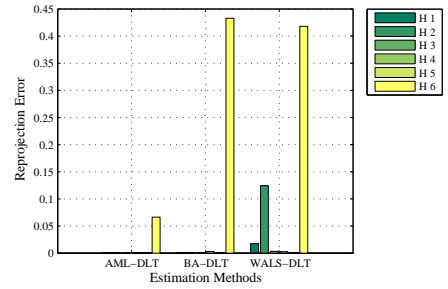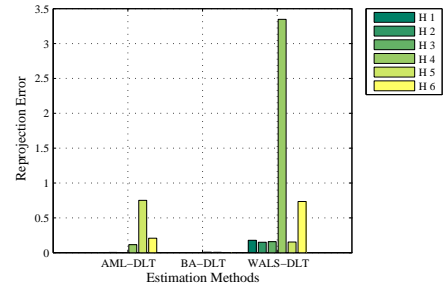
(j) view 6.
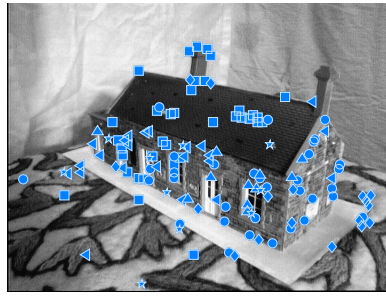
(k) view 7.

(l) Reprojection error for view 6 and 7.

(m) view 8.

(n) view 9.

(o) Reprojection error for view 8 and 9.

Continued Fig. 5: Comparison of consistency enforcing estimation methods on the Model House sequence. The first two columns show the feature points (inliers) associated with various homographies on the model house views. The third column compares the reprojection error of AML-DLT, WALS-DLT, and BA-DLT for each homography. Note that in some figures the errors associated with particular homographies are so close to zero that the bar graphs are barely visible.

where $\{\{\mathbf{m}_{n,i}, \mathbf{m}'_{n,i}\}_{n=1}^{N_i}\}_{i=1}^{I}$ are the original corresponding points and $\{\{\underline{\mathbf{m}}_{n,i}\}_{n=1}^{N_i}\}_{i=1}^{I}$ are arbitrary 2D points over which the minimum is taken. By plotting the error for each homography separately, it is possible to gain a deeper insight into the performance of each method. A comparison on real data is given in Figs. 5, 6, and 7. The results show that WALS is inferior to both AML and BA, frequently displaying much higher errors in at least one of the homographies. On the other hand, our proposed scheme produces results which are often a very good approximation to BA.

We defer a deeper discussion of the results to Section 8.

7.6 Application: robust consistent homography estimation

Our experiments have shown that the scope of consistency enforcing algorithms is not limited to synthetic data—when the algorithms are applied to homographies estimated on real-world scenes, full consistency can still be achieved. However, neither AML, nor WALS, nor BA are truly robust to outliers and it may be the case that if consistency constraints are imposed on a collection of homographies where one of the homographies is an outlier, then *all* of the methods may fail to produce accurate estimates. That is because each method employs a non-robust error measure and relies on an initialisation which in its current form is not robust.

One possible way to circumvent any robustness issues is to incorporate the consistency constraints directly into the well known random sampling and consensus (RANSAC) estimation loop [24, Sect. 4.7.1], so that the homographies returned by a RANSAC procedure are *by construction* fully consistent. This can be achieved by a simple modification of the canonical RANSAC procedure, and the pseudo-code for the modification is given in Algorithm 2.

Algorithm 2 makes use of several functions, namely f, g, h, and ρ. The function f can be any standard method for estimating a single homography, such as DLT or FNS. The function g serves to determine a set of inliers consistent with a given homography estimate. It utilises a cost function ρ that computes an error between a point correspondence and a homography, and declares a particular correspondence to be a member of the set of inliers for the given homography estimate if the value of ρ for that correspondence falls below a user-specified threshold t. Typically, the cost function ρ is chosen to be the symmetric transfer error. The function h represents a procedure that takes as input a set of homography matrices together with covariances and produces a new set of fully consistent homographies. In our implementation, we take for h the AML algorithm proposed in this paper.
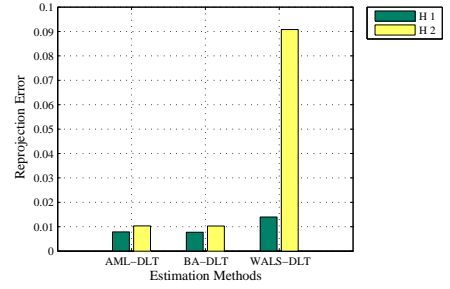
When faced with multiple planar structures in a set of correspondences, researchers and practitioners using stan-
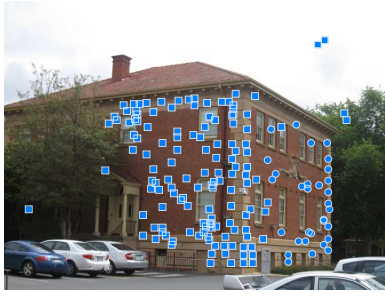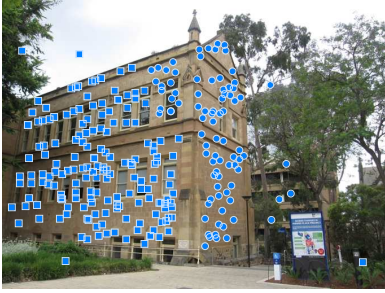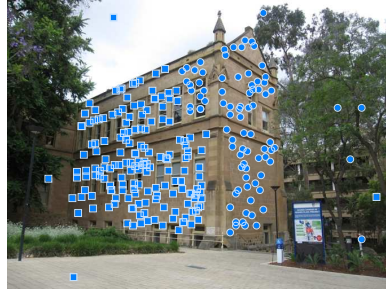
(a) Lady Symon building view 0.
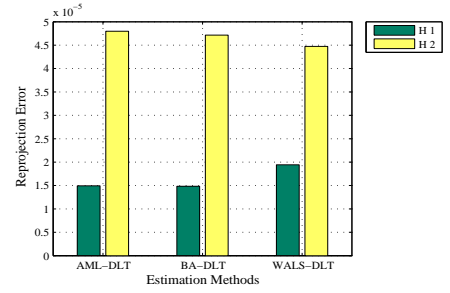

(b) Lady Symon building view 1.

(c) Reprojection error for view 0 and 1 of Lady Symon building.

(d) Old Classics Wing building view 0.

(e) Old Classics Wing building view 1.

(f) Reprojection error for view 0 and 1 of Old Classics Wing building.

Fig. 6: Comparison of consistency enforcing estimation methods on the AdelaideRMF sequence. The first two columns show the feature points (inliers) associated with various homographies on the Lady Symon and Old Classics Wing buildings. The third column compares the reprojection error of AML-DLT, WALS-DLT, and BA-DLT for each homography.

dard RANSAC frequently resort to the following strategy: (i) search for the homography with the largest number of inliers and (ii) from all considered homographies, choose the homography with the largest number of inliers, remove its inliers from the set of correspondences and repeat the whole process to find the next plane [29,44]. The essence of Algorithm 2 is the same, except that we modify step (ii) so that from all considered homographies, we choose the homography that has the largest number of inliers *and at the same time is fully consistent with any planar structures found so far*. We then remove its inliers from the set of correspondences and repeat the whole process to find the next plane.

Since computational efficiency is crucial for algorithms like RANSAC that are intended to be practical, our modification of RANSAC is an excellent example of where one would prefer to impose consistency constraints using our AML algorithm, which is relatively fast, instead of BA, which is slower. As a proof-of-concept of our idea, we show in Fig. 8 the inliers associated with *fully consistent* homographies on various real data taken from the Oxford dataset, which were produced by running our modified RANSAC algorithm. The results demonstrate that finding numerous visually pleasing inliers related to fully consistent homographies is fundamentally possible.

8 Discussion

We have identified several important factors that may help explain the success of our method. Firstly, the error covariance matrices associated with the individual homography estimates are absolutely crucial. Without covariance information we are unable to produce meaningful results. Secondly, our scale-invariant cost function captures the projective nature of the parameters that are being estimated, and ensures that the optimisation method always takes a step in a meaningful direction. The scale invariance is not properly accounted for in the method of Chen and Suter, and this partially explains why their method tends to produce worse results. Furthermore, the Levenberg–Marquardt optimisation method is better suited for solving non-linear optimisation problems than for example the bilinear approach followed by Chen and Suter.

Our modification of RANSAC serves to emphasise the practical utility of our research outcome. There are, of course, many other modifications to the original RANSAC algorithm, each claiming to be the best. Usually, the modifications are based on improvements to the sampling scheme. Although they may be faster than the original RANSAC, none can produce fully consistent homographies. We leave it to future work to decide which modern variant of RANSAC

(a) Traffic scene A view 0.

(b) Traffic scene A view 1.

(c) Reprojection error for view 0 and 1 of Traffic scene A.

(d) Traffic scene B view 0.

(e) Traffic scene B view 1.

(f) Reprojection error for view 0 and 1 of Traffic scene B.
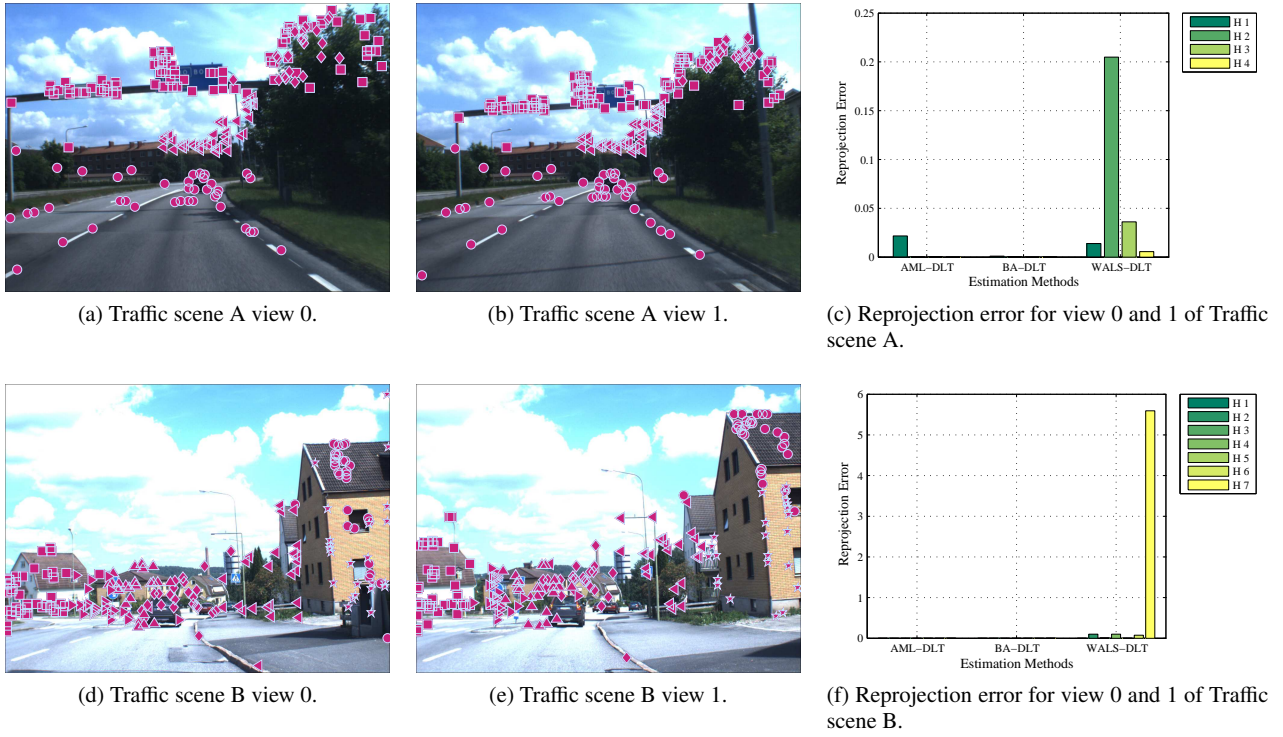
Fig. 7: Comparison of consistency enforcing estimation methods on the Traffic Signs sequence. The first two columns show the feature points (inliers) associated with various homographies on two traffic scenes. The third column compares the reprojection error of AML-DLT, WALS-DLT, and BA-DLT for each homography. Note that in some cases, the errors associated with particular homographies are so close to zero, that the bar graphs are barely visible.

(a) Feature points (inliers) associated with four fully consistent homographies on Merton College III.

(b) Feature points (inliers) associated with three fully consistent homographies on Raglan Castle.

(c) Feature points (inliers) associated with two fully consistent homographies on Valbonne Church.
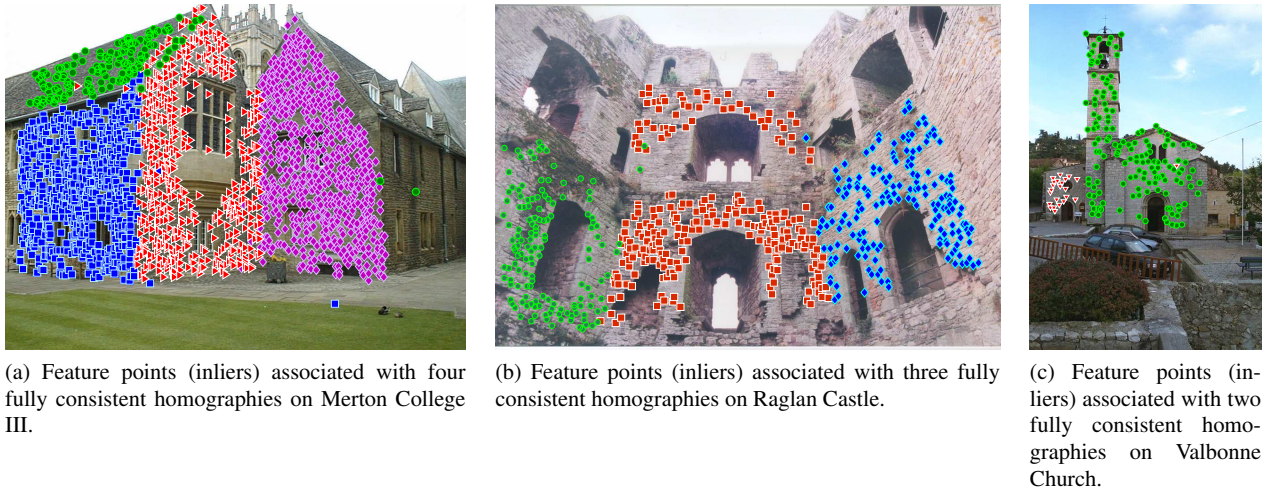
Fig. 8: Inliers associated with fully consistent homographies found using Algorithm 2. The results on real data indicate that finding fully consistent homographies using our AML cost function is tractable.

to couple with our full consistency constraints to achieve maximum computational efficiency.

We also wish to point out that our algorithm need not be incorporated into a sampling procedure such as RANSAC

to be of practical use. One could, for example, utilise the recently proposed generalised projection based M-estimator [36] to estimate a set of inconsistent homographies, and then upgrade them to a fully consistent set using our method.

Algorithm 2: ROBUST-CONSISTENT-HOMOGRAPHY-ESTIMATION

Input:
- $S = \{\mathbf{m}_n, \mathbf{m}'_n\}_{n=1}^N$ — a set of N corresponding pairs of points
- m — the minimum number of point correspondences needed to compute a homography
- $f\colon s \mapsto \mathbf{X}$ — the function estimating a homography matrix $\mathbf{X}$ from a sample s of m point correspondences
- I — the user-specified number of desired homographies
- $\rho(\mathbf{X}, \{\mathbf{m}, \mathbf{m}'\})$ — the cost function measuring the discrepancy between a point correspondence $\{\mathbf{m}, \mathbf{m}'\}$ and a homography $\mathbf{X}$
- t — the threshold for determining inliers
- $g\colon (\mathbf{X}, \rho, S, t) \mapsto \kappa$ — the function identifying a set of inliers κ within a set of data points S relative to a homography $\mathbf{X}$, based on a cost function ρ and a threshold t for determining inliers
- $h\colon ([\mathbf{X}_1, \ldots, \mathbf{X}_j], [\mathbf{\Lambda}_{\mathbf{x}_1}, \ldots, \mathbf{\Lambda}_{\mathbf{x}_j}]) \to [\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_j]$ — the function h that takes a collection of homographies $\mathbf{X}_1, \ldots, \mathbf{X}_j$ together with their covariances $\mathbf{\Lambda}_{\mathbf{x}_1}, \ldots, \mathbf{\Lambda}_{\mathbf{x}_j}$ and produces a new set of fully consistent homographies $\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_j$
- L — the maximum number of iterations (it is best to select L adaptively—see [24, Sect. 4.7.1] for details)

Output: $[\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_I]$ — fully consistent homographies

1 $[\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_I] \leftarrow \emptyset$ and $[\mathbf{\Lambda}_{\theta_1}, \ldots, \mathbf{\Lambda}_{\theta_I}] \leftarrow \emptyset$
2 **for** $i \leftarrow 1$ **to** I **do**
3 $\quad$ $k \leftarrow 0, C^* \leftarrow \infty, \mathbf{\Theta}_i^* \leftarrow \emptyset$, and $\kappa_i^* \leftarrow \emptyset$
4 $\quad$ **repeat**
5 $\quad\quad$ select a subset s^k of S of m point correspondences at random
6 $\quad\quad$ estimate parameters $\mathbf{X}^k = f(s^k)$ and compute covariance $\mathbf{\Lambda}_{\mathbf{x}^k}$
7 $\quad\quad$ compute cost $C^k = \sum_{\{\mathbf{m}, \mathbf{m}'\} \in S} \min(t, \rho(\mathbf{X}^k, \{\mathbf{m}, \mathbf{m}'\}))$
8 $\quad\quad$ **if** $C^* > C^k$ **then**
$\quad\quad\quad$ // Enforce consistency constraints on a candidate homography
9 $\quad\quad\quad$ $[\hat{\mathbf{\Theta}}_1, \ldots, \hat{\mathbf{\Theta}}_{i-1}, \hat{\mathbf{\Theta}}_i^k] \leftarrow h([\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_{i-1}, \mathbf{X}^k], [\mathbf{\Lambda}_{\theta_1}, \ldots, \mathbf{\Lambda}_{\theta_{i-1}}, \mathbf{\Lambda}_{\mathbf{x}^k}])$
10 $\quad\quad\quad$ compute cost $C^{k*} = \sum_{\{\mathbf{m}, \mathbf{m}'\} \in S} \min(t, \rho(\hat{\mathbf{\Theta}}_i^k, \{\mathbf{m}, \mathbf{m}'\}))$
11 $\quad\quad\quad$ compute inliers $\kappa_i^{k*} \leftarrow g(\hat{\mathbf{\Theta}}_i^k, \rho, S, t)$
12 $\quad\quad\quad$ **if** $|\kappa_i^*| < |\kappa_i^{k*}|$ **then**
13 $\quad\quad\quad\quad$ $C^* \leftarrow C^{k*}, \mathbf{\Theta}_i^* \leftarrow \hat{\mathbf{\Theta}}_i^k, \mathbf{\Lambda}_{\theta_i^*} \leftarrow \mathbf{\Lambda}_{\mathbf{x}^k}$, and $\kappa_i^* \leftarrow \kappa_i^{k*}$
14 $\quad\quad\quad$ **end if**
15 $\quad\quad$ **end if**
16 $\quad\quad$ $k \leftarrow k + 1$
17 $\quad$ **until** $k = L$
$\quad$ // Enforce consistency constraints on selected homographies
18 $\quad$ $[\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_{i-1}, \mathbf{\Theta}_i] \leftarrow h([\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_{i-1}, \mathbf{\Theta}_i^*], [\mathbf{\Lambda}_{\theta_1}, \ldots, \mathbf{\Lambda}_{\theta_{i-1}}, \mathbf{\Lambda}_{\theta_i^*}])$
19 $\quad$ $\mathbf{\Lambda}_{\theta_i} \leftarrow \mathbf{\Lambda}_{\theta_i^*}$ and $S \leftarrow S \setminus \kappa_i^*$
20 **end for**

Since enforcing consistency constraints leads to a reduction in reprojection error and hence an improvement in the quality of the homographies, it always makes sense to do so.

9 Conclusion

Our principal contribution within this paper has been to formulate the problem of estimating multiple interdependent homographies in a manner that enforces *full consistency* between all constituent homographies. A key feature of this formulation is the exploitation of latent variables that link together different homographies. The resulting solution strategy has a natural translation into a full-blown BA procedure and into the computationally more efficient AML estimation procedure. Since both BA and AML are non-linear optimisation techniques, they both require a suitable initialisation to produce meaningful results.

In this connection, our second important contribution has been the derivation of a novel compact initialisation pro-

cedure which can be applied when there are two or more planes in the scene. In contrast, the initialisation procedure of Chen and Suter is more involved and requires at least there planes before it can be utilised.

The third notable contribution of our work has been to show that our scale-invariant AML cost function frequently produces estimates that are an accurate approximation of BA. Compared to the WALS algorithm of Chen and Suter, our method has the advantage that it consistently converges to a better minimum. This is evidenced by experimental results on both real and synthetic data. Furthermore, we have furnished a derivation of suitable homography covariance matrices for both DLT and FNS estimates, to be found in Appendices A and B, and our final formulas are tailored to facilitate easy implementation.

Drawing on the experimentally confirmed validity of our approach, we have proposed a modification to the celebrated RANSAC algorithm to robustly estimate fully consistent homographies. Since in RANSAC computational efficiency is

vital, the modified algorithm serves as a noteworthy example of where our AML scheme is preferable to BA.

Finally, we conclude by pointing out that once recovered, the latent variables encoding the fully consistent homographies can be immediately utilised to provide a *projective reconstruction* of the scene.

The source code for our experiments can be found at `http://sites.google.com/site/szpakz/`.

Acknowledgement

Appendices

A Appendix: covariance of the AML estimate

Here, we derive a formula for the covariance matrix of the AML estimate of a vectorised homography matrix based on a set of image correspondences. It will be convenient to establish first an expression for the covariance matrix of the AML estimate of a parameter vector of a certain general model. This model will comprise, as particular cases, models whose parameters describe a relationship among image feature locations. Once the general formula for a covariance matrix is established, we shall then evolve a specialised formula for the case of the homography model.

A.1 General model

The data–parameter relationship for the general model will be assumed in the form

$$\mathbf{f}(\mathbf{z}, \boldsymbol{\beta}) = \mathbf{0},$$

where $\mathbf{z}$ is a length-k vector describing an ideal (noiseless) data point, $\boldsymbol{\beta}$ is a length-l vector of parameters, and $\mathbf{f}(\mathbf{z}, \boldsymbol{\beta})$ is a length-m vector of constraints of the form

$$\mathbf{f}(\mathbf{z}, \boldsymbol{\beta}) = \mathbf{U}(\mathbf{z})^\top \boldsymbol{\beta},$$

where $\mathbf{U}(\mathbf{z})$ is an $l \times m$ matrix—the so-called *data carrier matrix*—with entries formed by smooth functions in $\mathbf{z}$. Details on how this formulation applies to the homography model are given in Appendix A.2. It will be further assumed that the observed data points $\mathbf{z}_1, \dots, \mathbf{z}_N$ come equipped with covariance matrices $\boldsymbol{\Lambda}_{\mathbf{z}_1}, \dots, \boldsymbol{\Lambda}_{\mathbf{z}_N}$ quantifying measurement errors in the data. Under the assumption that the errors are independently sampled from Gaussian distributions with covariances of the form $\boldsymbol{\Lambda}_{\mathbf{z}_n}$, $n = 1, \dots, N$, the relevant AML

cost function to fit the model parameters to the data is given by

$$J_{\mathrm{AML}}(\boldsymbol{\beta}) = \sum_{n=1}^{N} \mathbf{f}(\mathbf{z}, \boldsymbol{\beta})^\top \boldsymbol{\Sigma}(\mathbf{z}_n, \boldsymbol{\beta})^{-1} \mathbf{f}(\mathbf{z}, \boldsymbol{\beta}),$$

where

$$\boldsymbol{\Sigma}(\mathbf{z}_n, \boldsymbol{\beta}) = \partial_{\mathbf{z}} \mathbf{f}(\mathbf{z}_n, \boldsymbol{\beta}) \boldsymbol{\Lambda}_{\mathbf{z}_n} [\partial_{\mathbf{z}} \mathbf{f}(\mathbf{z}_n, \boldsymbol{\beta})]^\top$$

(cf. [8, 10, 11, 26, 31, 35]). Importantly, when m, the common length of the $\mathbf{f}(\mathbf{z}_n, \boldsymbol{\beta})$'s, surpasses the codimension r of the submanifolds of the form $\{\mathbf{z} \in \mathbb{R}^k \mid \mathbf{f}(\mathbf{z}, \boldsymbol{\beta}) = \mathbf{0}\}$ with $\boldsymbol{\beta}$ representing parameters under which the data might have been generated, the inverses $\boldsymbol{\Sigma}(\mathbf{z}_n, \boldsymbol{\beta})^{-1}$ in the above expression for J_{AML} must be replaced by, say, the *r-truncated pseudo-inverses* $\boldsymbol{\Sigma}(\mathbf{z}_n, \boldsymbol{\beta})_r^+$ [17, 26]. Recall that the r-truncated pseudo-inverse of an $m \times m$ matrix $\mathbf{A}$, $\mathbf{A}_r^+$, is defined as follows: If $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ is SVD of $\mathbf{A}$, with $\mathbf{D} = \mathrm{diag}(d_1, \dots, d_m)$, and if $\mathbf{A}_r = \mathbf{U}\mathbf{D}_r\mathbf{V}^\top$ with $\mathbf{D}_r = \mathrm{diag}(d_1, \dots, d_r, 0, \dots, 0)$ is the r-truncated SVD of $\mathbf{A}$, then $\mathbf{A}_r^+ = \mathbf{V}\mathbf{D}_r^+\mathbf{U}^\top$ with $\mathbf{D}_r^+ = \mathrm{diag}(d_1^+, \dots, d_r^+, 0, \dots, 0)$, where $d_i^+ = d_i^{-1}$ when $d_i \neq 0$ and $d_i^+ = 0$ otherwise. The AML estimate of $\boldsymbol{\beta}$, $\widehat{\boldsymbol{\beta}}_{\mathrm{AML}}$, is the minimiser of J_{AML}. As a consequence of J_{AML} being homogeneous of degree zero, $\widehat{\boldsymbol{\beta}}_{\mathrm{AML}}$ is determined only up to scale. The estimate $\widehat{\boldsymbol{\beta}}_{\mathrm{AML}}$ satisfies the necessary optimality condition

$$[\partial_{\boldsymbol{\beta}} J_{\mathrm{AML}}(\boldsymbol{\beta})]_{\boldsymbol{\beta} = \widehat{\boldsymbol{\beta}}_{\mathrm{AML}}} = \mathbf{0}^\top, \tag{18}$$

which is the basis for all what follows. Using the formula

$$\mathbf{U}(\mathbf{z})^\top \boldsymbol{\beta} = (\mathbf{I}_m \otimes \boldsymbol{\beta}^\top) \mathrm{vec}(\mathbf{U}(\mathbf{z})), \tag{19}$$

one readily verifies that

$$[\partial_{\boldsymbol{\beta}} J_{\mathrm{AML}}(\boldsymbol{\beta})]^\top = 2\mathbf{X}_{\boldsymbol{\beta}} \boldsymbol{\beta},$$

where $\mathbf{X}_{\boldsymbol{\beta}} = \mathbf{M}_{\boldsymbol{\beta}} - \mathbf{N}_{\boldsymbol{\beta}}$ is an $l \times l$ symmetric matrix with

$$\mathbf{M}_{\boldsymbol{\beta}} = \sum_{n=1}^{N} \mathbf{U}_n \boldsymbol{\Sigma}_n^{-1} \mathbf{U}_n^T, \tag{20a}$$

$$\mathbf{N}_{\boldsymbol{\beta}} = \sum_{n=1}^{N} (\boldsymbol{\eta}_n^\top \otimes \mathbf{I}_l) \mathbf{B}_n (\boldsymbol{\eta}_n \otimes \mathbf{I}_l), \tag{20b}$$

$$\mathbf{U}_n = \mathbf{U}(\mathbf{z}_n), \tag{20c}$$

$$\mathbf{B}_n = \partial_{\mathbf{z}_n} \mathrm{vec}(\mathbf{U}_n) \boldsymbol{\Lambda}_{\mathbf{z}_n} [\partial_{\mathbf{z}_n} \mathrm{vec}(\mathbf{U}_n)]^\top, \tag{20d}$$

$$\boldsymbol{\Sigma}_n = (\mathbf{I}_m \otimes \boldsymbol{\beta}^\top) \mathbf{B}_n (\mathbf{I}_m \otimes \boldsymbol{\beta}), \tag{20e}$$

$$\boldsymbol{\eta}_n = \boldsymbol{\Sigma}_n^{-1} \mathbf{U}_n^\top \boldsymbol{\beta}. \tag{20f}$$

Accordingly, Eq. (18) can be rewritten as

$$\mathbf{X}_{\hat{\boldsymbol{\beta}}} \hat{\boldsymbol{\beta}} = \mathbf{0}, \tag{21}$$

where $\widehat{\boldsymbol{\beta}}_{\mathrm{AML}}$ is abbreviated to $\hat{\boldsymbol{\beta}}$ for clarity. Hereafter, $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}(\mathbf{z}_1, \dots, \mathbf{z}_N)$ will be assumed normalised and smooth as a function of $\mathbf{z}_1, \dots, \mathbf{z}_N$.

To derive an expression for the covariance matrix of $\hat{\boldsymbol{\beta}}$, we use (21) in conjunction with the covariance propagation formula

$$\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}} = \sum_{n=1}^{N} \partial_{\mathbf{z}_n}\hat{\boldsymbol{\beta}} \boldsymbol{\Lambda}_{\mathbf{z}_n}(\partial_{\mathbf{z}_n}\hat{\boldsymbol{\beta}})^{\top} \tag{22}$$

(cf. [16, 22]). Differentiating $\|\hat{\boldsymbol{\beta}}\|^2 = 1$ with respect to $\mathbf{z}_n$ gives $(\partial_{\mathbf{z}_n}\hat{\boldsymbol{\beta}})^{\top}\hat{\boldsymbol{\beta}} = \mathbf{0}$. This together with (22) implies that

$$\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}} = \boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}\hat{\boldsymbol{\beta}} = \mathbf{0}$$

so that $\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}$ is singular, and further yields

$$\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}} = \boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp} = \boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}, \tag{23}$$

where, of course, $\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp} = \mathbf{I}_l - \|\hat{\boldsymbol{\beta}}\|^{-2}\hat{\boldsymbol{\beta}}\hat{\boldsymbol{\beta}}^{\top}$. Letting $\mathbf{z}_n = [z_{n1} \ldots z_{nk}]^{\top}$ and $\hat{\boldsymbol{\beta}} = [\widehat{\beta_1}, \ldots, \widehat{\beta_l}]^{\top}$, and differentiating (21) with respect to z_{ni}, we obtain

$$\Big[[\partial_{z_{ni}}\mathbf{X}_{\boldsymbol{\beta}}]_{\boldsymbol{\beta}=\hat{\boldsymbol{\beta}}} + \sum_{j=1}^{l}[\partial_{\beta_j}\mathbf{X}_{\boldsymbol{\beta}}]_{\boldsymbol{\beta}=\hat{\boldsymbol{\beta}}}\partial_{z_{ni}}\widehat{\beta_j}\Big]\hat{\boldsymbol{\beta}} + \mathbf{X}_{\hat{\boldsymbol{\beta}}}\partial_{z_{ni}}\hat{\boldsymbol{\beta}} = \mathbf{0}.$$

Introducing the Gauss-Newton approximation, i.e., neglecting the terms that contain $\hat{\boldsymbol{\beta}}^{\top}\mathbf{u}(\mathbf{z}_n)$, we reduce this equality to the equality

$$\mathbf{U}_n\boldsymbol{\Sigma}_n^{-1}(\partial_{z_{ni}}\mathbf{U}_n)^{T}\hat{\boldsymbol{\beta}} + \mathbf{M}_{\hat{\boldsymbol{\beta}}}\partial_{z_{ni}}\hat{\boldsymbol{\beta}} = \mathbf{0}.$$

Now, in view of (19) and the fact that

$$(\partial_{z_{ni}}\mathbf{U}_n)^{T}\hat{\boldsymbol{\beta}} = \mathrm{vec}((\partial_{z_{ni}}\mathbf{U}_n)^{T}\hat{\boldsymbol{\beta}}) = \mathrm{vec}(\hat{\boldsymbol{\beta}}^{T}\partial_{z_{ni}}\mathbf{U}_n)$$
$$= (\mathbf{I}_m \otimes \hat{\boldsymbol{\beta}}^{\top})\partial_{z_{ni}}\mathrm{vec}(\mathbf{U}_n),$$

we have

$$\mathbf{M}_{\hat{\boldsymbol{\beta}}}\partial_{z_{ni}}\hat{\boldsymbol{\beta}} = -\mathbf{U}_n\boldsymbol{\Sigma}_n^{-1}(\partial_{z_{ni}}\mathbf{U}_n)^{T}\hat{\boldsymbol{\beta}}$$
$$= -\mathbf{U}_n\boldsymbol{\Sigma}_n^{-1}(\mathbf{I}_m \otimes \hat{\boldsymbol{\beta}}^{\top})\partial_{z_{ni}}\mathrm{vec}(\mathbf{U}_n)$$

and further

$$\mathbf{M}_{\hat{\boldsymbol{\beta}}}\partial_{\mathbf{z}_n}\hat{\boldsymbol{\beta}} = -\mathbf{U}_n\boldsymbol{\Sigma}_n^{-1}(\mathbf{I}_m \otimes \hat{\boldsymbol{\beta}}^{\top})\partial_{\mathbf{z}_n}\mathrm{vec}(\mathbf{U}_n).$$

Hence

$$\mathbf{M}_{\hat{\boldsymbol{\beta}}}\partial_{\mathbf{z}_n}\hat{\boldsymbol{\beta}}\boldsymbol{\Lambda}_{\mathbf{z}_n}(\partial_{\mathbf{z}_n}\hat{\boldsymbol{\beta}})^{\top}\mathbf{M}_{\hat{\boldsymbol{\beta}}}$$
$$= \mathbf{U}_n\boldsymbol{\Sigma}_n^{-1}(\mathbf{I}_m \otimes \hat{\boldsymbol{\beta}}^{\top})$$
$$\times \partial_{\mathbf{z}_n}\mathrm{vec}(\mathbf{U}_n)\boldsymbol{\Lambda}_{\mathbf{z}_n}[\partial_{\mathbf{z}_n}\mathrm{vec}(\mathbf{U}_n)]^{\top}(\mathbf{I}_m \otimes \hat{\boldsymbol{\beta}})\boldsymbol{\Sigma}_n^{-1}\mathbf{U}_n^{\top}.$$

But, by (20d) and (20e),

$$(\mathbf{I}_m \otimes \hat{\boldsymbol{\beta}}^{\top})\partial_{\mathbf{z}_n}\mathrm{vec}(\mathbf{U}_n)\boldsymbol{\Lambda}_{\mathbf{z}_n}[\partial_{\mathbf{z}_n}\mathrm{vec}(\mathbf{U}_n)]^{\top}(\mathbf{I}_m \otimes \hat{\boldsymbol{\beta}})$$
$$= (\mathbf{I}_m \otimes \hat{\boldsymbol{\beta}}^{\top})\mathbf{B}_n(\mathbf{I}_m \otimes \hat{\boldsymbol{\beta}}) = \boldsymbol{\Sigma}_n$$

so

$$\mathbf{M}_{\hat{\boldsymbol{\beta}}}\partial_{\mathbf{z}_n}\hat{\boldsymbol{\beta}}\boldsymbol{\Lambda}_{\mathbf{z}_n}(\partial_{\mathbf{z}_n}\hat{\boldsymbol{\beta}})^{\top}\mathbf{M}_{\hat{\boldsymbol{\beta}}} = \mathbf{U}_n\boldsymbol{\Sigma}_n^{-1}\boldsymbol{\Sigma}_n\boldsymbol{\Sigma}_n^{-1}\mathbf{U}_n^{\top}$$
$$= \mathbf{U}_n\boldsymbol{\Sigma}_n^{-1}\mathbf{U}_n^{\top}.$$

Therefore, in view of (20a),

$$\mathbf{M}_{\hat{\boldsymbol{\beta}}}\Big[\sum_{n=1}^{N} \partial_{\mathbf{z}_n}\hat{\boldsymbol{\beta}}\boldsymbol{\Lambda}_{\mathbf{z}_n}(\partial_{\mathbf{z}_n}\hat{\boldsymbol{\beta}})^{\top}\Big]\mathbf{M}_{\hat{\boldsymbol{\beta}}} = \sum_{n=1}^{N} \mathbf{U}_n\boldsymbol{\Sigma}_n^{-1}\mathbf{U}_n^{\top} = \mathbf{M}_{\hat{\boldsymbol{\beta}}},$$

or equivalently, on account of (22),

$$\mathbf{M}_{\hat{\boldsymbol{\beta}}}\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}\mathbf{M}_{\hat{\boldsymbol{\beta}}} = \mathbf{M}_{\hat{\boldsymbol{\beta}}}. \tag{24}$$

At this stage, one might be tempted to conclude that $\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}} = \mathbf{M}_{\hat{\boldsymbol{\beta}}}^{-1}$, but this would contravene the fact that $\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}$ is singular. To exploit (24) properly, we first note that, in view of (23),

$$\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp} = \boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}, \tag{25}$$

so we can rewrite (24) as

$$\mathbf{M}_{\hat{\boldsymbol{\beta}}}\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}\mathbf{M}_{\hat{\boldsymbol{\beta}}} = \mathbf{M}_{\hat{\boldsymbol{\beta}}}.$$

Pre- and post-multiplying the last equation by $\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}$ and letting

$$\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp} = \mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}\mathbf{M}_{\hat{\boldsymbol{\beta}}}\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}$$

yield

$$\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp}\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp} = \mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp}.$$

Pre- and post-multiplying this equation by $(\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+}$ further yields

$$(\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+}\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp}\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp}(\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+} = (\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+}\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp}(\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+}. \tag{26}$$

The matrix $\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp}$ is symmetric and its null space is, generically, spanned by $\hat{\boldsymbol{\beta}}$, so

$$\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp}(\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+} = (\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+}\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp} = \mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}$$

(cf. [1, Cor. 3.5]). We also have $(\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+}\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp}(\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+} = (\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+}$ by virtue of one of the four defining properties of the pseudo-inverse [1, Thm. 3.9]. Therefore (26) can be restated as

$$\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}}\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp} = (\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+},$$

which, on account of (25), implies

$$\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}} = (\mathbf{M}_{\hat{\boldsymbol{\beta}}}^{\perp})^{+}. \tag{27}$$

We now derive an alternate formula for the covariance matrix of $\hat{\boldsymbol{\beta}}$, namely

$$\boldsymbol{\Lambda}_{\hat{\boldsymbol{\beta}}} = \mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}(\mathbf{M}_{\hat{\boldsymbol{\beta}}})_{l-1}^{+}\mathbf{P}_{\hat{\boldsymbol{\beta}}}^{\perp}. \tag{28}$$

In this form, $\mathbf{\Lambda}_{\hat{\beta}}$ is explicitly expressed as $\mathbf{\Lambda}_{\hat{\beta}} = \mathbf{P}_{\hat{\beta}}^{\perp} \mathbf{\Lambda}_{\hat{\beta}}^{0} \mathbf{P}_{\hat{\beta}}^{\perp}$, with the pre-covariance matrix $\mathbf{\Lambda}_{\hat{\beta}}^{0} = (\mathbf{M}_{\hat{\beta}})_{l-1}^{+}$. We start by noting that, in view of (21), $\hat{\beta}$ is in the null space $\mathcal{N}(\mathbf{X}_{\hat{\beta}})$ of $\mathbf{X}_{\hat{\beta}}$. Generically, we may assume that $\mathcal{N}(\mathbf{X}_{\hat{\beta}})$ is spanned by $\hat{\beta}$. As $\mathbf{X}_{\hat{\beta}}$ is symmetric, the column space of $\mathbf{X}_{\hat{\beta}}$ is equal to the orthogonal complement of $\mathcal{N}(\mathbf{X}_{\hat{\beta}})$. In particular, $\mathbf{X}_{\hat{\beta}}$ has rank $l-1$. This together with $\mathbf{X}_{\hat{\beta}}$ being equal to $\mathbf{M}_{\hat{\beta}}$ to a first-order approximation implies that $\mathbf{X}_{\hat{\beta}}$ is in fact approximately equal to the $(l-1)$-truncated SVD of $\mathbf{M}_{\hat{\beta}}$, $(\mathbf{M}_{\hat{\beta}})_{l-1}$. Since the function $\mathbf{A} \mapsto \mathbf{A}^{+}$ is continuous on the set of all $l \times l$ matrices of constant rank $l-1$ [21,34,37,41], we have, approximately,

$$\mathbf{X}_{\hat{\beta}}^{+} = (\mathbf{M}_{\hat{\beta}})_{l-1}^{+}.$$

Taking into account that $\mathbf{X}_{\hat{\beta}}^{+} = \mathbf{P}_{\hat{\beta}}^{\perp} \mathbf{X}_{\hat{\beta}}^{+} \mathbf{P}_{\hat{\beta}}^{\perp}$, which immediately follows from (21), we see that, again approximately,

$$\mathbf{X}_{\hat{\beta}}^{+} = \mathbf{P}_{\hat{\beta}}^{\perp} (\mathbf{M}_{\hat{\beta}})_{l-1}^{+} \mathbf{P}_{\hat{\beta}}^{\perp}. \tag{29}$$

As a consequence of $\mathbf{M}_{\hat{\beta}}$ being approximately equal to $\mathbf{X}_{\hat{\beta}}$, $\mathbf{M}_{\hat{\beta}}^{\perp}$ ($= \mathbf{P}_{\hat{\beta}}^{\perp} \mathbf{M}_{\hat{\beta}} \mathbf{P}_{\hat{\beta}}^{\perp}$) is approximately equal to $\mathbf{P}_{\hat{\beta}}^{\perp} \mathbf{X}_{\hat{\beta}} \mathbf{P}_{\hat{\beta}}^{\perp} = \mathbf{X}_{\hat{\beta}}$. Both $\mathbf{M}_{\hat{\beta}}^{\perp}$ and $\mathbf{X}_{\hat{\beta}}$ have rank $l-1$, so their pseudo-inverses are also approximately equal,

$$(\mathbf{M}_{\hat{\beta}}^{\perp})^{+} = \mathbf{X}_{\hat{\beta}}^{+},$$

by the aforementioned continuity property of the pseudo-inverse. Hence, (29) can be restated as

$$(\mathbf{M}_{\hat{\beta}}^{\perp})^{+} = \mathbf{P}_{\hat{\beta}}^{\perp} (\mathbf{M}_{\hat{\beta}})_{l-1}^{+} \mathbf{P}_{\hat{\beta}}^{\perp},$$

and this in combination with (27) yields (28).

In the case that the matrices $\mathbf{\Sigma}(\mathbf{z}_n, \beta)^{-1}$ are replaced by the matrices $\mathbf{\Sigma}(\mathbf{z}_n, \beta)_r^{+}$ in the expression for J_{AML}, a similar change also affects the matrices $\mathbf{M}_{\hat{\beta}}$, $\mathbf{N}_{\hat{\beta}}$, and $\mathbf{X}_{\hat{\beta}}$. With $\mathbf{M}_{\hat{\beta}}$ suitably modified, formulae (27) and (28) continue to hold.

A.2 Homography model

If a planar homography is represented by an invertible 3×3 matrix $\mathbf{H}$ and if $\mathbf{m}' = [u', v', 1]^{\top}$ is the image of $\mathbf{m} = [u, v, 1]^{\top}$ by that homography, then

$$\mathbf{m}' \simeq \mathbf{H}\mathbf{m},$$

where $\simeq$ denotes equality up to scale. This relation can equivalently be written as

$$[\mathbf{m}']_{\times} \mathbf{H}\mathbf{m} = \mathbf{0}. \tag{30}$$

With $\beta = \text{vec}(\mathbf{H})$, $\mathbf{z} = [u, v, u', v']^{\top}$, and $\mathbf{U}(\mathbf{z}) = -\mathbf{m} \otimes [\mathbf{m}']_{\times}$, we have

$$[\mathbf{m}']_{\times} \mathbf{H}\mathbf{m} = \mathbf{U}(\mathbf{z})^{\top} \beta,$$

and so (30) can be restated as

$$\mathbf{U}(\mathbf{z})^{\top} \beta = \mathbf{0}. \tag{31}$$

The last relation encapsulates the homography model (for image motion) in the form conforming to the framework of Appendix A.1. Since the 9×3 matrix $\mathbf{U}(\mathbf{z})$ has rank 2, the three equations in (31) are linearly dependent and can be reduced—by deleting any one of them—to a system of two equations. For $\beta \neq \mathbf{0}$, the reduced system gives two functionally independent constraints on $\mathbf{z}$, and this has the consequence that the set of image correspondences $\{\mathbf{z} \in \mathbb{R}^4 \mid \mathbf{U}(\mathbf{z})^{\top} \beta = \mathbf{0}\}$ is a submanifold of $\mathbb{R}^4$ of codimension 2.

Let $\{\mathbf{m}_n, \mathbf{m}'_n\}_{n=1}^{N}$ be a set of image correspondences based on which an AML estimate of a homography is to be evolved. For each $n = 1, \ldots, N$, write $\mathbf{m}_n = [u_n, v_n, 1]^{\top}$ and $\mathbf{m}'_n = [u'_n, v'_n, 1]^{\top}$ and let $\mathbf{z}_n = [u_n, v_n, u'_n, v'_n]^{\top}$. Suppose that each pair $\mathbf{m}_n, \mathbf{m}'_n$ comes equipped with a pair of 2×2 respective covariance matrices $\mathbf{\Lambda}_{u_n,v_n}$, $\mathbf{\Lambda}_{u'_n,v'_n}$. For each $n = 1, \ldots, N$, let

$$\mathbf{\Lambda}_{\mathbf{z_n}} = \begin{bmatrix} \mathbf{\Lambda}_{u_n,v_n} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_{u'_n,v'_n} \end{bmatrix}.$$

Since $\{\mathbf{z} \in \mathbb{R}^4 \mid \mathbf{U}(\mathbf{z})^{\top} \beta = \mathbf{0}\}$ has codimension 2, the appropriate AML cost function is given by

$$J_{\text{AML}}(\beta) = \sum_{n=1}^{N} \beta^{\top} \mathbf{U}(\mathbf{z}_n) [\mathbf{\Sigma}(\mathbf{z}_n, \beta)]_2^{+} \mathbf{U}(\mathbf{z}_n)^{\top} \beta,$$

where

$$\mathbf{\Sigma}(\mathbf{z}_n, \beta) = (\mathbf{I}_3 \otimes \beta^{\top}) \mathbf{B}(\mathbf{z}_n) (\mathbf{I}_3 \otimes \beta),$$
$$\mathbf{B}(\mathbf{z}_n) = [\partial_{\mathbf{z}} \text{vec}(\mathbf{U}(\mathbf{z}))]_{\mathbf{z}=\mathbf{z}_n} \mathbf{\Lambda}_{\mathbf{z_n}} [[\partial_{\mathbf{z}} \text{vec}(\mathbf{U}(\mathbf{z}))]_{\mathbf{z}=\mathbf{z}_n}]^{\top},$$

and, explicitly,

$$\partial_{\mathbf{z}} \text{vec}(\mathbf{U}(\mathbf{z})) = -[\text{vec}(\mathbf{e}_1 \otimes [\mathbf{m}']_{\times}), \text{vec}(\mathbf{e}_2 \otimes [\mathbf{m}']_{\times}),$$
$$\text{vec}(\mathbf{m} \otimes [\mathbf{e}_1]_{\times}), \text{vec}(\mathbf{m} \otimes [\mathbf{e}_2]_{\times})],$$

with $\mathbf{e}_1 = [1, 0, 0]^{\top}$ and $\mathbf{e}_2 = [0, 1, 0]^{\top}$. Now, on account of (28), the covariance matrix of the AML estimate $\widehat{\beta}_{\text{AML}} = \text{vec}(\widehat{\mathbf{H}}_{\text{AML}})$ can be explicitly expressed as

$$\mathbf{\Lambda}_{\widehat{\beta}_{\text{AML}}} = \mathbf{P}_{\widehat{\beta}_{\text{AML}}}^{\perp} \mathbf{\Lambda}_{\widehat{\beta}_{\text{AML}}}^{0} \mathbf{P}_{\widehat{\beta}_{\text{AML}}}^{\perp},$$

where the pre-covariance matrix $\mathbf{\Lambda}_{\widehat{\beta}_{\text{AML}}}^{0}$ is given by

$$\mathbf{\Lambda}_{\widehat{\beta}_{\text{AML}}}^{0} = (\mathbf{M}_{\widehat{\beta}_{\text{AML}}})_8^{+},$$

$$\mathbf{M}_{\widehat{\beta}_{\text{AML}}} = \|\widehat{\beta}_{\text{AML}}\|^2 \sum_{n=1}^{N} \mathbf{U}(\mathbf{z}_n) [\mathbf{\Sigma}(\mathbf{z}_n, \widehat{\beta}_{\text{AML}})]_2^{+} \mathbf{U}(\mathbf{z}_n)^{\top}.$$

B Appendix: covariance of the DLT estimate

We finally derive the formula for the covariance matrix of the DLT estimate of a vectorised homography matrix under the assumption that the estimate is evolved from a normalised image data set.

Let $\mathbf{T}$ and $\mathbf{T}'$ be two transformations for normalising the coordinates of 2D image points,

$$\tilde{\mathbf{m}} = \mathbf{T}\mathbf{m} \quad \text{and} \quad \tilde{\mathbf{m}}' = \mathbf{T}'\mathbf{m}'.$$

The maps $\mathbf{T}$ and $\mathbf{T}'$ induce the corresponding transformation of homographies given by

$$\tilde{\mathbf{H}} = \mathbf{T}'\mathbf{H}\mathbf{T}^{-1}.$$

A defining characteristic of this latter transformation is that $\mathbf{m}' \simeq \mathbf{H}\mathbf{m}$ holds precisely when $\tilde{\mathbf{m}}' \simeq \tilde{\mathbf{H}}\tilde{\mathbf{m}}$. With $\boldsymbol{\beta} = \mathrm{vec}(\mathbf{H})$ and $\tilde{\boldsymbol{\beta}} = \mathrm{vec}(\tilde{\mathbf{H}})$, the transformation of homographies becomes

$$\tilde{\boldsymbol{\beta}} = (\mathbf{T}^{-\top} \otimes \mathbf{T}')\boldsymbol{\beta}.$$

Let $\{\tilde{\mathbf{m}}_n, \tilde{\mathbf{m}}'_n\}_{n=1}^N$ be a set of corresponding normalised 2D points. Set $\tilde{\mathbf{z}}_n = [\tilde{u}_n, \tilde{v}_n, \tilde{u}'_n, \tilde{v}'_n]^\top$ for each pair $\tilde{\mathbf{m}}_n = [\tilde{u}_n, \tilde{v}_n, 1]^\top$ and $\tilde{\mathbf{m}}'_n = [\tilde{u}'_n, \tilde{v}'_n, 1]^\top$, and let

$$\tilde{\mathbf{A}} = \sum_{n=1}^N \mathbf{U}(\tilde{\mathbf{z}}_n)\mathbf{U}(\tilde{\mathbf{z}}_n)^\top.$$

The DLT estimate of $\tilde{\boldsymbol{\beta}}$, $\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}$, based on $\{\tilde{\mathbf{m}}_n, \tilde{\mathbf{m}}'_n\}_{n=1}^N$ is defined as the minimiser $\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}$ of the cost function

$$\tilde{J}_{\mathrm{DLT}}(\tilde{\boldsymbol{\beta}}) = \frac{\tilde{\boldsymbol{\beta}}^\top \tilde{\mathbf{A}} \tilde{\boldsymbol{\beta}}}{\|\tilde{\boldsymbol{\beta}}\|^2}$$

and coincides with the eigenvector of $\tilde{\mathbf{A}}$ corresponding to the smallest value. The function $\tilde{J}_{\mathrm{DLT}}$ is similar in form to the function J_{AML}—the scalar quantity $\|\boldsymbol{\beta}\|^2$ plays in $\tilde{J}_{\mathrm{DLT}}$ the role of the matrices $\boldsymbol{\Sigma}_n$ in J_{AML}. Exploiting this observation, one can immediately put forward an argument along the lines of Appendix A.1, showing that $\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}$ has the covariance matrix in the form

$$\boldsymbol{\Lambda}_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}} = \mathbf{P}^\perp_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}} \boldsymbol{\Lambda}^0_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}} \mathbf{P}^\perp_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}},$$

where the pre-covariance matrix $\boldsymbol{\Lambda}^0_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}}$ is given by

$$\boldsymbol{\Lambda}^0_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}} = (\mathbf{M}_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}})_8^+ \mathbf{D}_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}} (\mathbf{M}_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}})_8^+,$$

$$\mathbf{D}_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}} = \|\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}^\top\|^{-2} \sum_{n=1}^N \mathbf{U}(\tilde{\mathbf{z}}_n)\boldsymbol{\Sigma}(\tilde{\mathbf{z}}_n, \widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}})\mathbf{U}(\tilde{\mathbf{z}}_n)^\top.$$

The details of the calculation leading to the above expression for $\boldsymbol{\Lambda}_{\widehat{\tilde{\boldsymbol{\beta}}}_{\mathrm{DLT}}}$, analogous to those presented in Appendix A.1, are omitted.

References

1. Albert, A.: Regression and the Moore–Penrose Pseudoinverse. Academic Press, New York (1972)
2. Baker, S., Datta, A., Kanade, T.: Parameterizing homographies. Tech. Rep. CMU-RI-TR-06-11, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA (2006)
3. Chen, P., Suter, D.: Error analysis in homography estimation by first order approximation tools: a general technique. J. Math. Imaging Vision **33**(3), 281–295 (2009)
4. Chen, P., Suter, D.: Rank constraints for homographies over two views: revisiting the rank four constraint. Int. J. Computer Vision **81**(2), 205–225 (2009)
5. Chin, T.J., Wang, H., Suter, D.: The ordered residual kernel for robust motion subspace clustering. In: Adv. Neural Inf. Process. Systems, vol. 22, pp. 333–341 (2009)
6. Chin, T.J., Yu, J., Suter, D.: Accelerated hypothesis generation for multistructure data via preference analysis. IEEE Trans. Pattern Anal. Mach. Intell. **34**(4), 625–638 (2012)
7. Chojnacki, W., Brooks, M.J.: On the consistency of the normalized eight-point algorithm. J. Math. Imaging Vision **28**(1), 19–27 (2007)
8. Chojnacki, W., Brooks, M.J., van den Hengel, A., Gawley, D.: On the fitting of surfaces to data with covariances. IEEE Trans. Pattern Anal. Mach. Intell. **22**(11), 1294–1303 (2000)
9. Chojnacki, W., Brooks, M.J., van den Hengel, A., Gawley, D.: Revisiting Hartley's normalized eight-point algorithm. IEEE Trans. Pattern Anal. Mach. Intell. **25**(9), 1172–1177 (2003)
10. Chojnacki, W., Brooks, M.J., van den Hengel, A., Gawley, D.: From FNS to HEIV: A link between two vision parameter estimation methods. IEEE Trans. Pattern Anal. Mach. Intell. **26**(2), 264–268 (2004)
11. Chojnacki, W., Brooks, M.J., van den Hengel, A., Gawley, D.: FNS, CFNS and HEIV: A unifying approach. J. Math. Imaging Vision **23**(2), 175–183 (2005)
12. Chojnacki, W., van den Hengel, A.: A dimensionality result for multiple homography matrices. In: Proc. 13th Int. Conf. Computer Vision, pp. 2104–2109 (2011)
13. Chojnacki, W., van den Hengel, A.: On the dimension of the set of two-view multi-homography matrices. Complex Anal. Oper. Theory **7**(2), 465–484 (2013)
14. Chojnacki, W., Hill, R., van den Hengel, A., Brooks, M.J.: Multiprojective parameter estimation for sets of homogeneous matrices. In: Proc. Int. Conf. Digital Image Computing: Techniques and Applications, pp. 119–124 (2009)
15. Chojnacki, W., Szpak, Z., Brooks, M.J., van den Hengel, A.: Multiple homography estimation with full consistency constraints. In: Proc. Int. Conf. Digital Image Computing: Techniques and Applications, pp. 480–485 (2010)
16. Csurka, G., Zeller, C., Zhang, Z., Faugeras, O.D.: Characterizing the uncertainty of the fundamental matrix. Comput. Vis. Image Underst. **68**(1), 18–36 (1997)
17. Engl, H.W., Hanke, M., Neubauer, A.: Regularization of Inverse Problems. Kluwer, Dordrecht (1996)
18. Fouhey, D.F., Scharstein, D., Briggs, A.J.: Multiple plane detection in image pairs using J-linkage. In: Proc. 20th Int. Conf. Pattern Recognition, pp. 336–339 (2010)
19. Fusiello, A.: A matter of notation: Several uses of the Kronecker product in 3D computer vision. Pattern Recognition Lett. **28**(15), 2127–2132 (2007)
20. Gao, J., Kim, S.J., Brown, M.S.: Constructing image panoramas using dual-homography warping. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 49–56 (2011)
21. Golub, G.H., Pereyra, V.: The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate. SIAM J. Numer. Anal. **10**(2), 413–432 (1973)

22. Haralick, R.M.: Propagating covariance in computer vision. Int. J. Pattern Recogn. Artif. Intell. **10**(5), 561–572 (1996)
23. Hartley, R.: In defense of the eight-point algorithm. IEEE Trans. Pattern Anal. Mach. Intell. **19**(6), 580–593 (1997)
24. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press, Cambridge (2004)
25. Kähler, O., Denzler, J.: Rigid motion constraints for tracking planar objects. In: Proc. 29th DAGM Symposium, *Lecture Notes in Computer Science*, vol. 4713, pp. 102–111 (2007)
26. Kanatani, K.: Statistical Optimization for Geometric Computation: Theory and Practice. Elsevier, Amsterdam (1996)
27. Kanatani, K., Morris, D.D.: Gauges and gauge transformations for uncertainty description of geometric structure with indeterminacy. IEEE Trans. Inform. Theory **47**(5), 2017–2028 (2001)
28. Kanatani, K., Ohta, N., Kanazawa, Y.: Optimal homography computation with a reliability measure. IEICE Trans. Inf. Syst. **E83-D**(7), 1369–1374 (2000)
29. Kanazawa, Y., Kawakami, H.: Detection of planar regions with uncalibrated stereo using distribution of feature points. In: Proc. 15th British Machine Vision Conf., pp. 247–256 (2004)
30. Larsson, F., Felsberg, M.: Using Fourier descriptors and spatial models for traffic sign recognition. In: Proc. 17th Scand. Conf. Image Analysis, *Lecture Notes in Computer Science*, vol. 6688, pp. 238–249 (2011)
31. Leedan, Y., Meer, P.: Heteroscedastic regression in computer vision: problems with bilinear constraint. Int. J. Computer Vision **37**(2), 127–150 (2000)
32. Lütkepol, H.: Handbook of Matrices. John Wiley & Sons, Chichester (1996)
33. Ma, Y., Soatto, S., Košecká, J., Sastry, S.: An Invitation to 3-D Vision: From Images to Geometric Models, 2nd edn. Springer, New York (2005)
34. Magnus, J.R., Neudecker, H.: Matrix Differential Calculus with Applications in Statistics and Econometrics. John Wiley & Sons, Chichester (1988)
35. Matei, B., Meer, P.: Estimation of nonlinear errors-in-variables models for computer vision applications. IEEE Trans. Pattern Anal. Mach. Intell. **28**(10), 1537–1552 (2006)
36. Mittal, S., Anand, S., Meer, P.: Generalized projection-based M-estimator. IEEE Trans. Pattern Anal. Mach. Intell. **34**(12), 2351–2364 (2012)
37. Penrose, R.: A generalized inverse for matrices. Math. Proc. Cambridge Philos. Soc. **51**(3), 406–413 (1955)
38. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes in C. Cambridge University Press, Cambridge (1995)
39. Scoleri, T., Chojnacki, W., Brooks, M.J.: A multi-objective parameter estimator for image mosaicing. In: Proc. IEEE Int. Symposium Signal Processing and its Applications, vol. 2, pp. 551–554 (2005)
40. Shashua, A., Avidan, S.: The rank 4 constraint in multiple ($\geq$ 3) view geometry. In: Proc. 4th European Conf. Computer Vision, *Lecture Notes in Computer Science*, vol. 1065, pp. 196–206 (1996)
41. Stewart, G.W.: On the continuity of the generalized inverse. SIAM J. Appl. Math. **17**(1), 33–45 (1969)
42. Szpak, Z.L., Chojnacki, W., Eriksson, A., van den Hengel, A.: Sampson distance based joint estimation of multiple homographies with uncalibrated cameras. Comput. Vis. Image Underst. **125**, 200–213 (2014)
43. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment—a modern synthesis. In: Proc. Int. Workshop Vision Algorithms, *Lecture Notes in Computer Science*, vol. 1883, pp. 298–372 (1999)
44. Vincent, E., Laganiere, R.: Detecting planar homographies in an image pair. In: Proc. 2nd Int. Symposium Image and Signal Processing Analysis, pp. 182–187 (2001)
45. Wong, H.S., Chin, T.J., Yu, J., Suter, D.: Dynamic and hierarchical multi-structure geometric model fitting. In: Proc. 13th Int. Conf. Computer Vision, pp. 1044–1051 (2011)
46. Zelnik-Manor, L., Irani, M.: Multiview constraints on homographies. IEEE Trans. Pattern Anal. Mach. Intell. **24**(2), 214–223 (2002)