Int J Comput Vis (2009) 81: 205–225
DOI 10.1007/s11263-008-0167-z

Rank Constraints for Homographies over Two Views: Revisiting the Rank Four Constraint

Pei Chen · David Suter

Received: 15 February 2007 / Accepted: 4 August 2008 / Published online: 13 September 2008

Abstract It is well known that one can collect the coefficients of five (or more) homographies between two views into a large, rank deficient matrix. In principle, this implies that one can refine the accuracy of the estimates of the homography coefficients by exploiting the rank constraint. However, the standard rank-projection approach is impractical for two different reasons: it requires many homographies to even score a modest gain; and, secondly, correlations between the errors in the coefficients will lead to poor estimates.

In this paper we study these problems and provide solutions to each. Firstly, we show that the matrices of the homography coefficients can be recast into two parts, each consistent with ranks of only one. This immediately establishes the prospect of *realistically* (that is, with as few as only three or four homographies) exploiting the redundancies of the homographies over two views. We also tackle the remaining issue: correlated coefficients. We compare our approach with the "gold standard"; that is, non-linear bundle adjustment (initialized from the ground truth estimate—the ideal initialization). The results confirm our theory and show one can implement rank-constrained projection and come close to the gold standard in effectiveness. Indeed, our algorithm (by itself), or our algorithm further refined by a bundle adjustment stage; may be a practical algorithm: providing generally better results than the "standard" DLT (direct linear transformation) algorithm, *and even better than the bundle adjustment result with the DLT result as the starting point*. Our unoptimized version has roughly the same cost as bundle adjustment and yet can generally produce close to the "gold standard" estimate (as illustrated by comparison with bundle adjustment *initialized from the ground truth*).

Independent of the merits or otherwise of our algorithm, we have illuminated why the naive approach of direct rank-projection is relatively doomed to failure. Moreover, in revealing that there are further rank constraints, not previously known; we have added to the understanding of these issues, and this may pave the way for further improvements.

Keywords Homography · Rank constraint · First order perturbation

P. Chen (✉)
School of Information Science and Technology,
Sun Yat-sen University, Guangzhou, China
e-mail: chenpei75@yahoo.com

P. Chen
Shenzhen Institute of Advanced Integration Technology,
CAS/CUHK, Shenzhen, China

D. Suter
ARC Centre for Perceptive and Intelligent Machines in Complex Environments, Department of Electrical and Computer Systems Engineering, Monash University, Melbourne, Australia

1 Introduction

The homography is a projective linear mapping, a special case of which is the mapping between the images of corresponding points, when those are the image points of 3D planar "real world" points. Homographies have many applications in computer vision and photogrammetry: the transfer of points from one image to another (registration, construction of panoramas etc.), and the extraction of camera and planar patch relative poses (robotics, photogrammetry etc.). This paper is concerned with jointly estimating several homographies over two views by exploiting rank constraints in a way that extends/improves on the well established fact that homographies can be related by a rank four constraint (Shashua and Avidan 1996) (see Sect. 3).

Rank projecting noisy data[1], onto a low rank subspace as a form of "denoising" was studied in Chen and Suter (2006). Assuming i.i.d. Gaussian noise, the error, still residing in the low-rank approximation matrix, depends on the ratio between the degree of freedom of the rank constrained matrix and the degree of freedom of the matrix without enforcing the rank constraint. The latter is simply the number of coefficients of the matrix. The former depends on the matrix dimensions *and on the rank* (it is typically much lower in size).

Thus, if we exploit the above mentioned rank four constraint, *the best, on average, we can hope for* (as the number of homographies approaches ∞), is that the error in the homographies coefficients will be reduced by a factor of $\sqrt{\frac{4}{9}}$ (Chen and Suter 2006). That is, a one third (33%) reduction in the average error in the coefficients of the homographies can be gained. However, a large number of homographies are required to gain *any significant* denoising, let alone close to the asymptotic (infinite number of planes) value cited above. Moreover, to actually gain *any* denoising from such a recovered subspace, *more than four planes are required.* There are many image pairs where it would be difficult to find more than four planes (of any significant size).

The real situation is even worse than this. The refinement starts from the *estimated* homography parameters and, even if the image measurements that were used to estimate these parameters were corrupted with i.i.d. Gaussian noise, the same can not be said about the parameters themselves. That is, the homography estimation suffers from heteroscedastic noise (Leedan and Meer 2000): The heteroscedastic noise arises due to the linearization inherent in the direct linear transformation (DLT) algorithm (Hartley and Zisserman 2003).

Without tackling these issues, it is not possible to achieve even the modest gains reported in Chen and Suter (2006) (i.e., using real data rather than synthetic data), let alone come close to the theoretical 33% gain for large numbers of homographies with i.i.d. Gaussian noise in their coefficients.

The purpose of this paper is to investigate these issues. In order to find a useful approach for so few planes, we study the special structure of the homography. Thus in the first part of this paper, we are particularly interested in how to obtain rank constraints from the homographies of ≤ 3 planes. We can do this because the rank-four constraint is too general. The collection of homographies has even more structure. Starting from the fact that the homographies can be shown to lie in a rank-three *affine* subspace, we show how three homographies suffice to obtain useful constraints: as it turns out that only a subset of dimension-three affine subspaces are possible homographies. More importantly, we show that there is even more exploitable structure and that the homography coefficients can be shown to lie in the "combination" of two rank-one matrices and thus more constraints are available to reliably improve the homography accuracy, as few as three planes.

In order to overcome the difficulty, associated with the heteroscedastic noise in the homography parameters, we use estimates of the covariances, obtained by employing first order approximation techniques, leading to an effective weighted low-rank approximation.

Our solution is inherently iterative. Likewise, the "gold standard" approach of non-linear optimization (bundle adjustment) is also iterative. With iterative approaches there are two crucial issues—convergence and initialization. For convergence, we show that convergence (to a local minimum) is guaranteed. The issue of local vs global minimum is linked to the issue of initialization. We show, experimentally, that our approach generally converges to a very good solution, using at most two starting points, that we recommend as alternative initializations. Indeed, we show results that are very competitive with the "gold standard" (even when the latter is initialized with the ground truth—in general a better starting point than one would normally be able to use).

Related Work

We note that Zelnik-Manor and Irani (1999a, 1999b, 2002) have studied a related problem. They define the "relative homographies" of two planes over multiple views and show that these also reside in a dimension-four subspace but then go on to show that homologies can be scaled to reside in a dimension-three subspace. Furthermore, they show that the dimension four constraint also holds for the case of multiple-plane-over-multiple-view. Zelnik-Manor and Irani contrast their homology constraint with the homography constraint and emphasize the claimed advantage of the former in requiring only three planes rather than four (Zelnik-Manor and Irani 2002). A significant part of the contribution of our paper is that we show this is somewhat misleading as our homography rank-constraints over two views can indeed be applied to less than four planes. Indeed, although we do devise and implement an algorithm to demonstrate practical exploitation of rank-constraints with very few homographies: a major contribution is to expose and study the impediments to exploitation of rank-constrained homography denoising.

Rank constraints in computer vision tasks also arise in the factorization approach to the problem of structure from motion (SFM), under the assumption of an orthography (Tomasi and Kanade 1990, 1992). This includes the setting

[1] Generally, the measured data will not maintain the low rank predicted from the noise-free quantities.

of affine camera models (Poelman and Kanade 1994, 1997; Hartley and Zisserman 2003). The SFM problem can be reduced to a rank-one factorization problem, by using a reference frame (Aguiar and Moura 1999a, 2003). Face recognition is another prominent setting where rank constraints can be exploited: It was proved, by using spherical harmonics, that "all Lambertian reflectance functions obtained with arbitrary distant sources lie close to a 9D linear subspace" (Basri and Jacobs 1999, 2003; Ramamoorthi and Hanrahan 2001; Ramamoorthi 2002).

The standard approach to exploiting rank constraints involves computing an SVD. In order to overcome the limitations of batch-based methods, like the SVD, a sequential method was proposed to do the factorization (Morita and Kanade 1997). A similar technique was employed to deal with missing data problem with low rank matrix (Brand 2002).

Heteroscedastic noise, requires more sophisticated approaches: weighted factorization (Aguiar and Moura 2000, 2003) was proposed to deal with different-level noise; and directional noise was further investigated in Irani and Anandan (2000), Anandan and Irani (2002). In Aguiar and Moura (2000, 2003) and Anandan and Irani (2002), Irani and Anandan (2000), the noise was assumed to be frame independent or approximately frame independent. That is, in Aguiar and Moura (2000, 2003), the noise of one feature point is assumed to be i.i.d. Gaussian, across the frames (different feature points have a different noise level); and in Anandan and Irani (2002), Irani and Anandan (2000), the directional noise of one feature is characterized by its covariance matrix and this covariance matrix remains unchanged across the frames (different feature points have a different covariance matrix). In Irani and Anandan (2000), Anandan and Irani (2002), Irani and Anandan showed how to approximately reduce frame-dependent directional noise to be frame-independent.

Organization of the Paper

In Sect. 2, we briefly review the rank four constraint. In Sect. 3, we show how to exploit further structure with homographies over two views. In Sect. 4, we first review the (normalized) DLT algorithm for homography estimation, and then propose how to jointly estimate ≥ 3 homographies. In Sect. 5, simulations and experiments are presented.

2 Rank-Four Constraint

First, we cite the *Result 12.1* on p. 312 of Hartley and Zisserman (2003), which describes the relationship between a homography and the related projection matrices. Suppose the projection matrices for two views are $\mathbf{P} = [\mathbf{I}|\mathbf{0}]$ and $\mathbf{P}' = [\mathbf{R}|\mathbf{t}]$. The ith plane is defined by $\pi_i^T \mathbf{X} = 0$, with $[\pi_i]_{4\times 1} = [-\mathbf{v}_i^T \ 1]^T$[2] and $[\mathbf{X}]_{4\times 1}$ as the homogeneous representation of 3D points on the plane. The homography induced by the plane is, with a matrix representation:

$$\mathcal{H}_i \sim \mathbf{R} + \mathbf{t}\mathbf{v}_i^T \tag{1}$$

Thus, with the knowledge of $\mathbf{R}$ and $\mathbf{t}$,[3] the homography of the ith plane is characterized by the vector $\mathbf{v}_i$. Note that this is a *particular* representation (we call it the *canonical representation*): all matrices related to this matrix by a scale are also representations of the same homography. In some applications, we need to relate the matrix homography $\mathcal{H}_i$ to its vector form $\mathbf{h}_i$.[4] Suppose

$$\mathcal{H}_i = \begin{bmatrix} h_{1,i} & h_{2,i} & h_{3,i} \\ h_{4,i} & h_{5,i} & h_{6,i} \\ h_{7,i} & h_{8,i} & h_{9,i} \end{bmatrix}$$

$\mathbf{h}_i$ is defined to be $\mathbf{h}_i = [h_{1,i} \ h_{2,i} \ \ldots \ h_{9,i}]^T$.

The matrix $\mathbf{H} = [\mathbf{h}_1 \ \mathbf{h}_2 \ \ldots \ \mathbf{h}_n]_{9\times n}$, whose columns are homographies in canonical form, can be expressed as the following, in terms of $\mathbf{R}$, $\mathbf{t}$, and $\{\mathbf{v}_i\}$:

$$\begin{aligned}[\mathbf{H}]_{9\times n} = {} & [vec(\mathbf{R}^T)]_{9\times 1}[1 \quad 1 \quad \ldots \quad 1]_{1\times n} \\ & + [\mathbf{U_t}]_{9\times 3}[\mathbf{v}_1 \quad \mathbf{v}_2 \quad \ldots \quad \mathbf{v}_n]_{3\times n}\end{aligned} \tag{2}$$

where $vec(\mathbf{M})$ is the column-first vectorization form of matrix $\mathbf{M}$, and

$$[\mathbf{U_t}]_{9\times 3} = \begin{bmatrix} t_1\mathbf{I}_3 \\ t_2\mathbf{I}_3 \\ t_3\mathbf{I}_3 \end{bmatrix}_{9\times 3} \tag{3}$$

Homographies, calculated according to (1) (with the knowledge of $\mathbf{R}$, $\mathbf{t}$ and $\{\mathbf{v}_i\}$), are embedded in a dimension-three affine subspace. However, homographies are only defined up to a scale factor, and in practice we estimate the homographies from image measurements (not directly from $\mathbf{R}$,

[2] Here, we use $\pi_i = [-\mathbf{v}_i^T \ 1]^T$, instead of $\pi_i = [\mathbf{v}_i^T \ 1]^T$ (Hartley and Zisserman 2003). Consequently, $\mathcal{H}_i \sim \mathbf{R} + \mathbf{t}\mathbf{v}_i^T$ in (1), not $\mathcal{H}_i \sim \mathbf{R} - \mathbf{t}\mathbf{v}_i^T$.

[3] Here, and throughout this paper, we use the notations of $\mathbf{R}$ and $\mathbf{t}$. For calibrated images these are the rotation and translations of the camera. However, everything works also for uncalibrated images where $\mathbf{R}$ and $\mathbf{t}$ are then related to the rotation and translation through the 3×3 invertible calibration matrix. For the calibrated case, where we refer to the fundamental matrix, one should replace it with the essential matrix.

[4] In this paper, a homography will be represented either as a 3×3 matrix or as a 9×1 vector. We will use $\mathcal{H}_i$ or $\mathbf{h}_i$ for a matrix homography or its vectorized form, respectively. When its form can be determined from the context, we only use the term of "homography", which may be a matrix or a vector. $\mathbf{H}$ will denote the matrix that have several vector homographies as its columns, as defined below. $\mathbf{H}$ is referred to as "homography matrix".

t and $\{\mathbf{v}_i\}$), and thus we know the homography up to an unknown scale (and perturbed with noise). We can choose to select a special matrix representation for the class of matrices representing the homography—for example, we usually normalize the homography so that $||\mathcal{H}_i||_{Frobenius} = 1$. However, any such choice will lead to different (and unknown) scale factors between the chosen representative and the homography matrix defined by (1). In effect, this means that each column of the homography matrix **H**, in practice, will be scaled up with a different (and unknown) factor. As a consequence, we can only calculate the dimension-four subspace, rather than the dimension-three affine subspace. However, the homography coefficients do lie in a particular restricted set of dimension-three affine manifolds and this fact can be exploited (as we show next). Thus, although Zelnik-Manor and Irani (1999b, 2002) emphasized the limitation of homography-rank constaints (in being only applicable to more than four planes in contrast to their homology based constraints), we demonstrate a way around this limitation: as we can demonstrate effective denoising with as few as three planes/homographies.

3 Calculation of the Dimension-Four Subspace

In this section, we show how to exploit the special structure of the homography from (1) and (2). First, we start with the well known dimension-four subspace from ≥ 4 homographies. We then show how to extract structure from less than four planes. **More importantly, we reveal a previously unreported fact that the homography matrix can be expressed as the "combination" of two rank-one matrices**. (Note, here, the term of "combination" has a special meaning, as will be explained in Sect. 3.3.) This fact can be employed to design an iterative algorithm for enforcing these quite stringent rank constraints.

3.1 Subspace Constraint from ≥ 4 Planes

This is the well established approach and it is included here for completeness.

Suppose that n (≥ 4) planes are observed over two views. The rank of the homography matrix **H** is four (Shashua and Avidan 1996), as shown above. In other words, all the vector homographies (columns of **H**) are restricted to a dimension four subspace.

All we need do, in principle, is to use the SVD (Golub and Loan 1996) to project the columns of **H** onto the basis formed by the left singular vectors associated with the four largest singular values. However, as mentioned above, this will not be optimal if the noise in the homography coefficients is heteroscedastic. Moreover, to achieve any reasonable "denoising" effect, much greater than four planes are needed in practice.

3.2 Subspace Constraint from ≥ 3 Planes

In this section, we first show it is possible to calculate the dimension-four subspace from only three homographies.

3.2.1 Calculation of the Direction of Translation **t**

The crux of this, and subsequent subsections, is that we concentrate on estimating the direction, of the camera pose translation **t**, then solving for the remaining structure.

From the structure of $[\mathbf{U_t}]_{9\times 3}$ in (3), one can see that only a restricted set of affine spaces are possible—regardless of the values of **t** there are directions in R^9 that are not in the span of the columns of $[\mathbf{U_t}]_{9\times 3}$. From (2), one can see that the dimension-3 subspace $[\mathbf{U_t}]_{9\times 3}$ is determined by the direction of **t**. In the following, we show that the direction of **t** can be calculated from only three homographies.

(a) *First, construct a* $3C_n^2 \times 6$ *matrix* **M** *from n homographies*.

From two homographies with matrix representations $\mathcal{H}_i$ and $\mathcal{H}_j$, a 3×6 matrix $\mathbf{M}_{i,j}$ is constructed as in Appendix A. For n ($n \geq 3$) homographies, we stack all the pairwise matrices $\mathbf{M}_{i,j}$ for $i < j$, obtaining a $3C_n^2 \times 6$ matrix $[\mathbf{M}]_{3C_n^2\times 6}$:

$$[\mathbf{M}]_{3C_n^2\times 6} = \begin{bmatrix}\mathbf{M}_{1,2}^T & \cdots & \mathbf{M}_{1,n}^T & \mathbf{M}_{2,3}^T & \cdots & \mathbf{M}_{2,n}^T & \cdots & \mathbf{M}_{n-1,n}^T\end{bmatrix}^T \tag{4}$$

with $\mathbf{M}_{i,j} \in R^{3,6}$ defined in Appendix A.

(b) *It can be shown that rank*$(\mathbf{M}) = 3$, *except in degenerate cases, see Appendix B*. Thus, except in these degenerate cases, the right null-space of **M** is of dimension 3.

(c) *Now we show how to calculate the vector* **t** *from matrix* **M** *in* (4).

Construct the 3×9 matrix $[\mathbf{S}]_{3\times 9}$ from the null subspace of **M**:

$$[\mathbf{S}]_{3\times 9} = [\mathcal{N}_1 \quad \mathcal{N}_2 \quad \mathcal{N}_3]_{3\times 9} \tag{5}$$

where $[\mathbf{N}]_{6\times 3}$ is the basis vectors spanning the null-space of **M** and each 6×1 column of **N** is arranged as a 3×3 matrix:

$$\mathcal{N}_i = \begin{bmatrix} n_{1,i} & n_{4,i} & n_{5,i} \\ n_{4,i} & n_{2,i} & n_{6,i} \\ n_{5,i} & n_{6,i} & n_{3,i} \end{bmatrix} \tag{6}$$

In the noise free situation, **S** is rank deficient and has a rank of two; and **t** is the left null vector of **S**, as will be proved in Appendix C. In practice, the left singular vector of **S**, associated with the least singular value, is taken as the solution of **t**.

So far, we have shown how to calculate **t** *from* ≥ 3 *homographies.*

In the analysis above, we need to calculate the dimension-three null subspace of $\mathbf{M}$. If the homographies are noise free, there exists such a null subspace. However, error is inevitably introduced to the homographies, due to the presence of noise in feature points. Faced with this, we can, in principle, use the SVD to calculate this null subspace: taking the three right singular vectors that are associated with the three smallest singular values.

However, heteroscedastic noise is introduced in the matrix $\mathbf{M}$. Suppose the noise in $\mathbf{M}$ has a covariance matrix $\mathbf{C_M}$. We first employ the bilinear approach (Chen 2004; Chen and Suter 2007) or the AP approach (Manton et al. 2003) to calculate the $\mathbf{C_M}$ weighted rank-three approximation matrix: $\mathbf{M}_{\mathbf{C}}^{3}$.[5] (Chen 2004; Chen and Suter 2007). Then, a reasonable solution of $\mathbf{N}$ is the null subspace of this $\mathbf{M}_{\mathbf{C}}^{3}$. The characterization of $\mathbf{C_M}$ is given in Appendix E.

3.2.2 *Calculation of the Fourth Basis Vector of the Dimension Four Subspace*

Without loss of generality, suppose $\|\mathbf{t}\|_{Frobenius} = 1$. From (2), we define $\mathbf{H}^{\dagger} = (\mathbf{I}_9 - \mathbf{U_t}\mathbf{U_t}^T)\mathbf{H}$ as the projection on the orthogonal complement of $span(\mathbf{U_t})$, where $\mathbf{U_t}$ is defined in (3). It can be easily seen that $rank(\mathbf{H}^{\dagger}) = 1$, because each column of $\mathbf{H}^{\dagger}$, $\mathbf{h}_i^{\dagger}$, is parallel to $(\mathbf{I}_9 - \mathbf{U_t}\mathbf{U_t}^T)vec(\mathbf{R}^T)$. The columns $\mathbf{h}_i^{\dagger}$ are different from each other up to an unknown scale, because the canonical form of homographies, as in (1), can not be calculated in practice. In the noise free case, the fourth vector of the dimension four subspace can be taken as any column of $\mathbf{H}^{\dagger}$. Note that the fourth vector calculated this way is not parallel to $vec(\mathbf{R}^T)$, but parallel to its orthogonal complement to the subspace spanned by $\mathbf{U_t}$.

3.3 "Combination" of Two Rank-One Matrices

In this subsection, we further study the special structure of the homography matrix: The new result is that we derive a constraint **expressed as the "combination" of two rank-one matrices.** The starting point is also (2). We rewrite this formula as

$$\mathbf{H} = \mathbf{H}' + \mathbf{H}'' \tag{7}$$

where $\mathbf{H}'$ and $\mathbf{H}''$ are the first and second parts of the right side of (2), respectively. Obviously, the former part has a rank of one, i.e., it can be factored as $\mathbf{H}' = \mathbf{u}\mathbf{d}^T$. As for the second part, we rearrange each column of $\mathbf{H}''$, $\mathbf{h}_i'' = [h_{1,i}'' \quad h_{2,i}'' \quad \dots \quad h_{9,i}'']^T$, as a 3×3 matrix

$$\mathcal{H}_i'' = \begin{bmatrix} h_{1,i}'' & h_{2,i}'' & h_{3,i}'' \\ h_{4,i}'' & h_{5,i}'' & h_{6,i}'' \\ h_{7,i}'' & h_{8,i}'' & h_{9,i}'' \end{bmatrix}$$

and juxtapose them into a $3 \times 3n$ matrix: $\breve{\mathbf{H}}'' = [\mathcal{H}_1'' \ \mathcal{H}_2'' \ \dots \ \mathcal{H}_n'']_{3\times 3n}$. Due to the fact that $\mathcal{H}_i'' = \mathbf{t}\mathbf{v}_i^T$, $[\breve{\mathbf{H}}'']_{3\times 3n}$ can be factored as $[\breve{\mathbf{H}}'']_{3\times 3n} = [\mathbf{t}]_{3\times 1}[\mathbf{v}_2^T \ \mathbf{v}_2^T \ \dots \ \mathbf{v}_n^T]_{1\times 3n} = \mathbf{t}\mathbf{v}^T$, then it is obviously a rank-one matrix.[6] For clarity, we define a function f for denoting the rearrangement of the matrix $\mathbf{H}''$ as $\breve{\mathbf{H}}''$, as $f : R^{9,n} \to R^{3,3n}$ with $f(\mathbf{H}'') = \breve{\mathbf{H}}''$. Similarly, define the inverse of f, $f^{-1} : R^{3,3n} \to R^{9,n}$ with $f^{-1}(\breve{\mathbf{H}}'') = \mathbf{H}''$.

We use the word of "combination", because $\mathbf{H}$ is not the direct sum of two rank-one matrices. It is the sum of two parts, each part can be written in rank-one form by the processes of re-ordering the coefficients as described above.

How can this property be utilized in the estimation of homographies? With the knowledge of $\mathbf{H}'$, we can calculate $\breve{\mathbf{H}}''$ as the rank-one approximation of $f(\mathbf{H} - \mathbf{H}')$. Vice versa. This suggests we iteratively update $\mathbf{H}'$ and $\mathbf{H}''$ while holding $\mathbf{H}''$ and $\mathbf{H}'$ constant, respectively. Further, these two steps can be turned into computing one of $\mathbf{u}$, $\mathbf{d}$, $\mathbf{t}$, and $\mathbf{v}$ while holding the other three constant.

Assuming the error in the homography matrix is characterized by the covariance matrix $[\mathbf{C_H}]_{9n\times 9n}$, the homography subspace estimation problem is to estimate $\mathbf{u}$, $\mathbf{d}$, $\mathbf{t}$, and $\mathbf{v}$, by minimizing the following $\mathbf{C_H}$ weighted objective function:

$$\|\mathbf{H} - \mathbf{u}\mathbf{d}^T - f^{-1}(\mathbf{t}\mathbf{v}^T)\|_{\mathbf{C}} \tag{8}$$

where the covariance matrix $\mathbf{C_H}$[7] is analyzed in Appendix D.

Thus we alternately compute one of $\mathbf{u}$, $\mathbf{d}$, $\mathbf{t}$ and $\mathbf{v}$:

$$\begin{aligned} \mathbf{u} &= \arg\min_{\mathbf{x}} \|\mathbf{H} - f^{-1}(\mathbf{t}\mathbf{v}^T) - \mathbf{x}\mathbf{d}^T\|_{\mathbf{C}} \\ \mathbf{d} &= \arg\min_{\mathbf{x}} \|\mathbf{H} - f^{-1}(\mathbf{t}\mathbf{v}^T) - \mathbf{u}\mathbf{x}^T\|_{\mathbf{C}} \\ \mathbf{t} &= \arg\min_{\mathbf{x}} \|f(\mathbf{H} - \mathbf{u}\mathbf{d}^T) - \mathbf{x}\mathbf{v}^T\|_{\mathbf{C}} \\ \mathbf{v} &= \arg\min_{\mathbf{x}} \|f(\mathbf{H} - \mathbf{u}\mathbf{d}^T) - \mathbf{t}\mathbf{x}^T\|_{\mathbf{C}} \end{aligned} \tag{9}$$

All of these four steps can be abstracted as the following $\mathbf{C_H}$ weighted minimization problem:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{B} - \mathbf{a}\mathbf{x}^T\|_{\mathbf{C}} \tag{10}$$

The solution of (10) can be found in Chen (2004), Chen and Suter (2007). Because in each step of (9), the objective function defined in (8) decreases, the iterations will converge to

[5] Here, for a concise notation, the subscript $\mathbf{M}$ of $\mathbf{C_M}$ is omitted. This also applies to $\mathbf{C_H}$ in (8–10).

[6] It is important to be aware that the reshaping of the coefficients does not *produce* exploitable rank constraints. A general $9 \times n$ matrix has just as many degrees of freedom as a general 3 by $3n$ matrix. The reshaping *exposes* a rank constraint that shows the coefficients have less than $9n$ degrees of freedom.

[7] In order for a simple equation, we use $\mathbf{C}$ instead of $\mathbf{C_H}$, for example in (9), when no ambiguity is resulted in.

a local minimum. An algorithm, based on the analysis here, will be presented as the ***Core algorithm*** in Sect. 4.2.

Note that, because $\mathbf{t}\mathbf{v}^T$ and $\mathbf{u}\mathbf{d}^T$ are two components of $\mathbf{H}$, they should be computed pair by pair, i.e., $\mathbf{t}$ and $\mathbf{v}$ should be computed one after another and this also applies to the pair of $\mathbf{u}$ and $\mathbf{d}$. Please also note that the order between $\mathbf{t}$ and $\mathbf{v}$ (or between $\mathbf{u}$ and $\mathbf{d}$) does not make any difference.

Before concluding this section, let us examine the degrees of freedom. First, we recall that a rank r $m \times n$ matrix has a degree of freedom of $mr + nr - r^2$ (Chen and Suter 2006). Since the $9 \times n$ homography matrix $\mathbf{H}$ is of rank four, its degree of freedom is $9 \times 4 + 4n - 4 \times 4$, i.e., $4n + 20$ for $n \geq 4$. From (7) that $\mathbf{H}$ can be decomposed as a rank-one $9 \times n$ matrix and a rank-one $3 \times 3n$ matrix, the degree of freedom of the homography matrix $\mathbf{H}$ is: $(9+n-1)+(3+3n-1)$, i.e. $4n+10$ for $n \geq 2$. There exists a difference of 10 degree of freedom, under these two constraints. This is significant for small n (e.g., for $n = 4$ the ratio is $\frac{26}{36} = 0.72$ indicating a potential average improvement of 28%). In terms of improving existing rank constraints, we note the similarity in what we have just achieved with homographies to the work of (Aguiar and Moura 1999a, 1999b, 2000, 2001, 2003) in improving rank-constrained structure from motion: reducing the rank four (or rank three) SFM problem to a rank-one problem.

Careful counting of the overall degrees of freedom shows that we still have not fully exploited all constraints and thus further improvement may be possible (but we have yet to discover how to expose and exploit such further reductions in the degrees of freedom).

4 Algorithms for Homography Refinement

The previous sections exposed possible ways to refine homography coefficients and hinted at algorithmic issues. In this section we provide detail on practical approaches.

4.1 DLT, Normalized DLT and Subspace Constrained DLT

We first review the normalized direct linear transform (DLT) algorithm (Hartley and Zisserman 2003) for homography estimation. For simplicity, we suppose the third coordinate of the homogeneous representation of a point is one.

Homography

$$\mathcal{H}_k = \begin{bmatrix} h_{1,k} & h_{2,k} & h_{3,k} \\ h_{4,k} & h_{5,k} & h_{6,k} \\ h_{7,k} & h_{8,k} & h_{9,k} \end{bmatrix}$$

maps points $\mathbf{x} = [x_1 \quad x_2 \quad 1]^T$ of the kth plane in the first view on $\mathbf{x}' = [x'_1 \quad x'_2 \quad 1]^T$ in the second view: $\mathbf{x}' \sim \mathcal{H}_k \mathbf{x}$.

From the cross product $\mathbf{x}' \times \mathcal{H}_k \mathbf{x} = \mathbf{0}$, each pair of the matches, $\{\mathbf{x}_i, \mathbf{x}'_i\}$, produces a 3×9 matrix:

$$\mathbf{A}_i = \begin{bmatrix} \mathbf{0} & -\mathbf{x}_i^T & x'_{2,i}\mathbf{x}_i^T \\ \mathbf{x}_i^T & \mathbf{0} & -x'_{1,i}\mathbf{x}_i^T \\ -x'_{2,i}\mathbf{x}_i^T & x'_{1,i}\mathbf{x}_i^T & \mathbf{0} \end{bmatrix}_{3\times 9} \tag{11}$$

which satisfies $\mathbf{A}_i \mathbf{h}_k = \mathbf{0}$. For p ($p \geq 4$) pairs of matches, stack all $\mathbf{A}_i$ ($1 \leq i \leq p$) as

$$[\mathbf{A}]_{3p\times 9} = [\mathbf{A}_1^T \quad \ldots \quad \mathbf{A}_p^T]^T \tag{12}$$

$\mathbf{A}\mathbf{h}_k = \mathbf{0}$ holds. $\mathbf{h}_k$ is taken as the right singular vector of $\mathbf{A}$, associated with the least singular value. This is the DLT algorithm (Hartley and Zisserman 2003) for homography estimation.

In Hartley and Zisserman (2003), a normalization step is recommended. It consists of a translation and a scaling, so that the centroid of the transformed points is the origin $(0, 0)$ and their average distance from the origin is $\sqrt{2}$. Suppose the centroid of the original points is (c_1, c_2) and their average distance to this centroid is l. The normalization transform $\mathbf{T}$ is

$$\begin{bmatrix} \frac{1}{l} & 0 & -\frac{c_1}{l} \\ 0 & \frac{1}{l} & -\frac{c_2}{l} \\ 0 & 0 & 1 \end{bmatrix} \tag{13}$$

Similarly, a normalization transform for the second view, $\mathbf{T}'$, can be calculated.

The normalized DLT algorithm takes the DLT algorithm as the core algorithm. First, calculate the transformed points for each view and their associated normalization transforms $\mathbf{T}$ and $\mathbf{T}'$. Second, using DLT, calculate the homography $\widetilde{\mathcal{H}}_k$ from the normalized matches. Finally, in the denormalization step, set

$$\mathcal{H}_k = \mathbf{T}'^{-1} \widetilde{\mathcal{H}}_k \mathbf{T} \tag{14}$$

as the homography for the plane in the original views.

When the dimension-four subspace is known, as $\mathbf{U} \in R^{9,4}$, the subspace constrained DLT solution is calculated the following way: First, calculate $\hat{\mathbf{x}}$ as the solution of

$$\mathbf{A}\mathbf{U}\mathbf{x} = \mathbf{0} \tag{15}$$

by standard smallest singular value way. Second, take the

$$\mathbf{h} = \mathbf{U}\hat{\mathbf{x}} \tag{16}$$

as the solution of the homography, which is obviously embedded in the subspace $\mathbf{U}$.

4.2 Algorithm for Jointly Estimating ≥ 3 Homographies

The algorithm presented here is based on the analysis in Sects. 3.2.1, 3.2.2 and 3.3.

Algorithm

1. Taking all the feature points in the n planes as a whole set, calculate the normalization transforms $\mathbf{T}$ and $\mathbf{T}'$, for the first view and the second view respectively.
2. Calculate the translation $\mathbf{t}$ for the normalized views.
   (a) Calculate each homography, $\mathbf{h}_i$, using the normalized DLT algorithm.
   (b) Using (4), calculate the matrix $\mathbf{M}$ from $\mathbf{h}_i$.
   (c) Calculate $\mathbf{N}$, the dimension-three null subspace of $\mathbf{M}$.
   (d) From $\mathbf{N}$, calculate the matrix $\mathbf{S}$, according to (5); and calculate $\mathbf{t}$ as the left null vector of $\mathbf{S}$.
3. From the normalized planes, calculate the dimension four subspace $\mathbf{U}$ of the homographies and refine using the rank-one parts. This step (the "***Core algorithm***") is explained further in the following. At the end of this step, we have improved estimates of $\mathbf{t}$ and $\mathbf{u}$ from which we immediately have a basis for the subspace.
4. For each normalized plane, calculate its subspace-$\mathbf{U}$ constrained homography, by (15) and (16).
5. Calculate the denormalized homographies for all the planes, as in the denormalization step (14) of the normalized DLT.

4.2.1 Core Algorithm

In this subsection, we concentrate on an iterative algorithm for step 3—we call this the "***Core algorithm***". This step is based on formula (9):

Core algorithm (with an initial estimate of $\mathbf{t} = \mathbf{t}^0$ as its input)

1. Initialize by calculating $\mathbf{u}^0$, $\mathbf{d}^0$ and $\mathbf{v}^0$, in turn. First, $\mathbf{u}^0$ and $\mathbf{d}^0$ are the solution of minimizing $\|\mathbf{u}^0(\mathbf{d}^0)^T - (\mathbf{I}_9 - \mathbf{U_t}\mathbf{U_t}^T)\mathbf{H}\|_{Frobenius}$, as can be solved by the SVD (Golub and Loan 1996). $\mathbf{t}^0$ is solved as the solution of minimizing $\|f(\mathbf{H} - \mathbf{u}^0(\mathbf{d}^0)^T) - \mathbf{t}^0\mathbf{v}^{0T}\|_{Frobenius}$, with f defined in Sect. 3.3.
2. Iterate until convergence: Compute $\mathbf{t}^{i+1}$, when holding $\mathbf{u}^i$, $\mathbf{d}^i$ and $\mathbf{v}^i$ constant. Ditto for computing $\mathbf{v}^{i+1}$, $\mathbf{u}^{i+1}$, and $\mathbf{d}^{i+1}$ in turn.

Note that we use superscripts to indicate iteration indices and that the vectors of $\mathbf{u}^i$, $\mathbf{d}^i$, $\mathbf{t}^i$ and $\mathbf{v}^i$ are estimates of those quantities (without superscripts) in Sect. 3.3.

We find that, in a few cases, the ***Core algorithm*** is trapped in a non-global minimum, measured by (8). We overcome this problem by starting from two different initial $\mathbf{t}^0$: One is the estimate used in the proposed ***algorithm*** and another is the left null vector of the associated Fundamental matrix. The subspace, which fits the homographies better, i.e., has a smaller objective function defined in (8), is selected as the solution.

5 Experimental Results

In this section, we show the gain of jointly estimating ≥ 3 homographies: compared with separately estimating each homography by using the normalized DLT algorithm. We also compare with the "gold-standard" bundle adjustment (BA) (Triggs et al. 2000).

The original BA algorithm tries to simultaneously estimate the projection matrices and 3D feature points so that they minimize the sum of the squared difference between the observed 2D feature points and their 2D reprojected positions (calculated from estimated projection matrices and estimated 3D feature points). Here, we use a variant of the BA algorithm for jointly estimating several homographies over two views. The starting point is also (1). The homographies can be determined from $\mathbf{R}$, $\mathbf{t}$, and $\{\mathbf{v}_i\}$:

$$\{\hat{\mathbf{R}}, \hat{\mathbf{t}}, \{\hat{\mathbf{v}}_i\}\} = \arg\min_{\mathbf{R},\mathbf{t},\{\mathbf{v}_i\}} \sum_{i=1}^{n}\sum_{j=1}^{p_i}\big(d(\mathbf{x}'_j, (\mathbf{R}+\mathbf{t}\mathbf{v}_i^T)\mathbf{x}_j)^2 + d(\mathbf{x}_j, (\mathbf{R}+\mathbf{t}\mathbf{v}_i^T)^{-1}\mathbf{x}'_j)^2\big) \tag{17}$$

where p_i is the number of feature points in the i^{th} plane, and $d(\mathbf{x}, \mathbf{y})$ denotes the Euclidean distance between inhomogeneous points represented by $\mathbf{x}$ and $\mathbf{y}$.

By using the Levenberg-Marquardt method (Press et al. 1992), the optimal solution of $\mathbf{R}$, $\mathbf{t}$, and $\{\mathbf{v}_i\}$ can be found, given an accurate initial starting point. However, we also find that, when the noise levels are different in different planes, the performance of the bundle adjustment deteriorates. In order to overcome this problem, we use:

$$\{\hat{\mathbf{R}}, \hat{\mathbf{t}}, \{\hat{\mathbf{v}}_i\}\} = \arg\min_{\mathbf{R},\mathbf{t},\{\mathbf{v}_i\}} \sum_{i=1}^{n}\sum_{j=1}^{p_i}\left(\frac{d(\mathbf{x}'_j, (\mathbf{R}+\mathbf{t}\mathbf{v}_i^T)\mathbf{x}_j)^2}{\sigma_i^2} + \frac{d(\mathbf{x}_j, (\mathbf{R}+\mathbf{t}\mathbf{v}_i^T)^{-1}\mathbf{x}'_j)^2}{\sigma_i^2}\right) \tag{18}$$

where σ_i is estimated from the DLT homographies: $\sigma_i = \sqrt{\frac{\sum_{j=1}^{p_i}(d(\mathbf{x}'_j, \hat{\mathcal{H}}_i\mathbf{x}_j)^2 + d(\mathbf{x}_j, \hat{\mathcal{H}}'_i\mathbf{x}'_j)^2)}{2(n_i-4)}}$ (and holding fixed during iteration).

We compare three algorithms for estimating the homographies: normalized Direct Linear Transform (referred to simply as **DLT**), the proposed algorithm (referred to as "proposed" and given the mnemonic "**pro**"), and Bundle Adjustment (referred to as **BA**). In actual fact, taking into account

the two possible objective functions, and the three possible initializations of **BA**, we have 6 variants of **BA**. Those using the objective of (17) have their mnemonic **1** as suffix, and those using (18) have mnemonic **0** as suffix. The three initializations are: with the ground truth of $\mathbf{R}$, $\mathbf{t}$, and $\{\mathbf{v}_i\}$ as the initial point (mnemonic **GT**), with the result from the proposed algorithm, and with the result of the DLT. In summary we have the following 8 combinations:

- **DLT**
- **proposed**
- variants of BA:
  - those using objective (18)
    - using DLT as starting point **BA-DLT-0**
    - using proposed algorithm as starting point **BA-pro-0**
    - using the ground truth as starting point **BA-GT-0**
  - those using objective (17)
    - using DLT as starting point **BA-DLT-1**
    - using proposed algorithm as starting point **BA-pro-1**
    - using the ground truth as starting point **BA-GT-1**

Of course, **BA-GT-0** and **BA-GT-1** are not practical variants of the algorithms. However, *they do provide an indication of, in some sense, the best one can do with bundle adjustment* as the issue of how to initialize within the basin of attraction of the true solution should be side-stepped. The variants **BA-DLT-0** and **BA-DLT-1** essentially represent current "state of the art" (and practical) yardsticks. As we will demonstrate, our algorithm provides a competitive result with these yardsticks and close to the best yardstick (BA starting from the ground truth). Furthermore, we show that by initializing BA from our algorithm (**BA-pro-0** and **BA-pro-1**), one does generally better than the current *practical* yardsticks **BA-DLT-0** and **BA-DLT-1**.

5.1 Validation Using Known Ground Truth

5.1.1 Synthetic Data

Experimental setting

- *Cameras* Suppose there exists a 3D coordinate system for the world. The first camera is randomly positioned on the $z = 0$ plane, at $(x, y, 0)$ that is uniformly distributed in $(0 \sim 3, 0 \sim 3, 0)$. The second camera is also placed on the plane of $z = 0$ at $(-x, -y, 0)$ with random perturbations up to 0.3 in x- and y- coordinates. The cameras randomly rotate with the constraint that both of their projection axes pass through the point $(0, 0, 40)$ on the z-axis. We assume pinhole models for the cameras, and that their image planes are assumed to be placed between the camera centers and the objects, with a unit distance from their camera centers.
- *Planes* The planes are determined in the following way: Each of them passes through a point on the z-axis: $(0, 0, \eth)$, where $\eth$ is a random value between $35 \sim 45$. The angles between z-axis and the planes are between $45 \sim 80$ degrees. For each plane, 20 points randomly distribute in $(-10 \sim 10, -10 \sim 10, *)$, where the star symbol "$*$" means that the z-coordinate is determined by the plane and the x- and y-coordinates. Note only those feature points in one plane that are not obscured by the other two planar segments can be selected.
- *Feature points* With the knowledge of both cameras and 3D feature points in the planes, we can obtain 2D feature points on two views. In order to make the simulation more realistic, the feature points on two views are scaled so that they are restricted in the region of $(-256 \sim 256, -256 \sim 256)$. A particular representative of the feature points in one view is shown in Fig. 1.
- *Noise* 0-mean i.i.d. Gaussian noise is added to different planes. In order to present an informative comparison for the methods, mentioned above, we include situations where the starting estimates of the homography of one plane is estimated with less accuracy than the others. We arbitrarily choose the third plane to be an estimate resulting from r times stronger noise. The different noise levels are chosen as σ (0.5, 1, 1.5, 2 and 2.5), and different ratio r as (1, 2, 3, 4 and 5).
- *Performance evaluation* In the following experiments/simulations, we use the reprojection error as the index for comparison. A commonly used index for evaluating the estimated homography $\hat{\mathcal{H}}$ is:

$$\sqrt{\frac{\sum_{i=1}^{p} d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + d(\mathbf{x}_i', \hat{\mathbf{x}}_i')^2}{2p}} \quad \text{subject to } \hat{\mathbf{x}}_i' \sim \hat{\mathcal{H}}\hat{\mathbf{x}}_i \quad \forall i \tag{19}$$

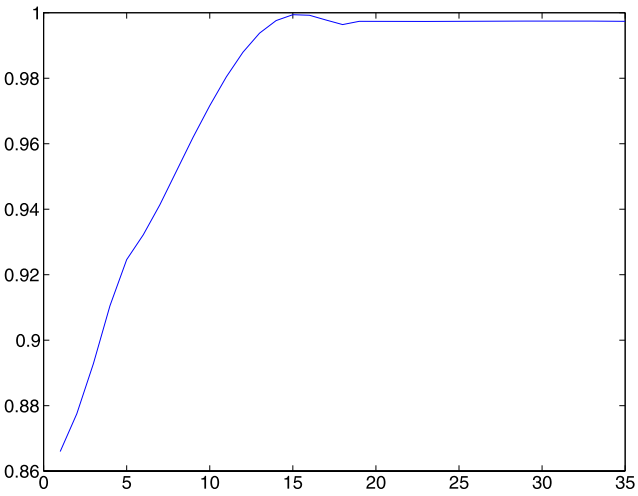


Fig. 1 The correlation coefficient between the direction of the estimate $\mathbf{t}$ and its ground-truth data $\tilde{\mathbf{t}}$

where $\hat{\mathbf{x}}$ denotes the estimate of $\mathbf{x}$. However, in all methods investigated here, we work only with homography coefficients and the algorithms do not compute estimates of feature points, $\{\hat{\mathbf{x}}\}$. In order to overcome this difficulty, we prefer to use the noise free data, if available, in comparison:

$$\sqrt{\frac{\sum_{i=1}^{p} d(\bar{\mathbf{x}}_i', \hat{\mathcal{H}}\bar{\mathbf{x}}_i)^2 + d(\bar{\mathbf{x}}_i, \hat{\mathcal{H}}'\bar{\mathbf{x}}_i')^2}{2p}} \tag{20}$$

where $\bar{\mathbf{x}}$ denotes for the ground truth of $\mathbf{x}$.

- *Termination* We terminate the iteration of the proposed algorithm (or the BA algorithms (17)–(18)) when the change of the objective function in (8) (or (17)–(18)) is less than 10^{-10}.
- *Statistics* In order to obtain enough data for the statistics, we repeat 100 times for every setting (with a same noise level σ and a same ratio of r).

In Fig. 2, the average error for the least noisy planes (first two planes) is shown: In each sub-figure, the abscissa is the ratio of the noise level of the third plane to that in the first two planes and the ordinate is the average error. In Fig. 3, the average error for the third plane is shown. The main conclusion is that the proposed algorithm (and BA of (18)) perform much better than the DLT algorithm. The proposed algorithm performs only slightly worse than the BA of (18).

We can also make several other side-observations. (a) As mentioned in the beginning of this section, the BA algorithm of (17) deteriorates when the noise levels in different planes are different. Although the BA algorithm of (17) has a smaller error than that of the DLT for the third plane (see Fig. 3), it is obtained at the price of increasing the error for the first two planes, when the ratio is 2 to 5 (see Fig. 2). The BA of (18) is generally better than the BA of (17). (b) When the noise is not stronger, for example from 0.5 to 1.5, the BA of (18), starting from the result from the proposed one algorithm, is better than that starting from the DLT.

For space limitations, we omit the results of our experiments using four or five planes but the same conclusions hold.

5.1.2 A Real Example: Model House

We use the Model House data (views, 2D and 3D points, and projection matrices are available at http://www.robots.ox.ac.uk/~vgg/data.html.), in particular the 8th and 9th views (see Fig. 4). There are 168 matches on the two views. From these 168 matches, four prominent planes are segmented into: the front of the house, the roof of the house, the ground, the side of the house, each with 50/51/18/37 matches: by the RANSAC algorithm (Fischler and Rolles 1981).

Because the 3D points are available, we can (as a first investigation) measure the error under controlled noise. The measured 3D points, can be said to be on some plane but only within some error tolerance. Thus, we first "purify" (least squares fit) the 3D feature points so that they are exactly on the appropriate plane. Then we project these "purified" 3D points using the projection matrices available, to obtain the data with ground truth. Different levels of noise is then added to these virtual feature points, as in Sect. 5.1.1 (in this case picking the plane of the side of the house to have the differing noise level).

The experimental results are shown in Figs. 5 and 6. We can draw essentially the same conclusions as those in Sect. 5.1.1 except that, in this particular example, only the samples of noise change from run to run, 2,500 runs in total—the underlying noise-free data remains unchanged. (Thus, the curves of BA-pro-0/BA-GT-0/BA-DLT-0 (and BA-pro-1/BA-GT-1/BA-DLT-1) merge into one in Figs. 5 and 6. It seems that this configuration of data produces a basin of attraction that includes all three starting points.)

5.1.3 Discussion

- It is noted (end of Sect. 4) that two initializations are used to produce a more reliable solution in the proposed algorithm in Sect. 4.2: the estimate $\mathbf{t}$ (derived by step 2 of Sect. 4.2) and the fundamental matrix null vector. Here, we give a numerical characterization of the reliability, in terms of the convergence rate to the *global* optimum.

  Actually, the global optimal solution of (8) is not known in practice. This difficulty can be overcome by utilizing the ground-truth data. The estimate, by starting from the noise-free data, is assumed to be the global minimum; and the estimates from the two initializations above are regarded to converge to the global minimum if their objective function after termination is near enough to the *global* minimum.

  It is found that the global minimum can be estimated in almost all cases, by jointly using the two different initializations above. In all $100 \times 5 \times 5 \times 3$ trials in Sects. 5.1.1, 5.1.2 (plus our synthetic planes experiments involving 4 planes, not detailed here) the global minimum is not found only in 24 cases. In terms of the two initializations separately, the global minimum is not found in 123/258 cases, respectively, by the two initializations mentioned above. (Note that the proposed algorithm converges in all cases, though with a few cases to non-global minimum.)
- Parameters for comparison? As noted in the experimental setting in Sect. 5.1.1, the reprojection error in (20) is used as the index for comparison. It may be thought that the parameters $\mathbf{R}$, $\mathbf{t}$ and $\{\mathbf{v}_i\}$ in (17) and (18) (or $\mathbf{u}$, $\mathbf{d}$, $\mathbf{t}$ and $\mathbf{v}$ in (8)) can be used as yardsticks as they have physical interpretations.

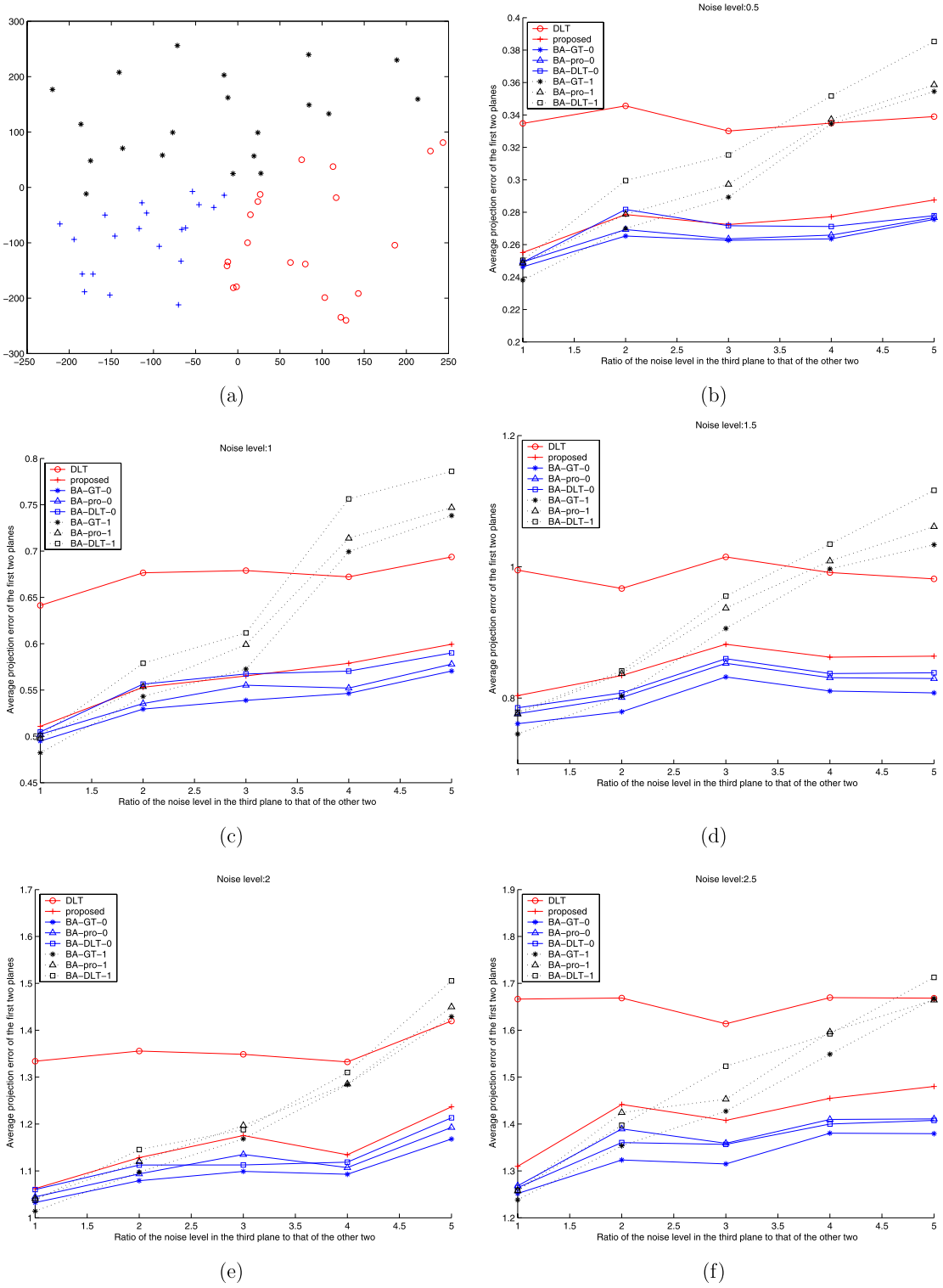


Fig. 2 Experimental result for 3-plane case: the first two planes. In **b–f**, the abscissa is the ratio of the noise level of the third plane to that in the first two planes and the ordinate is the average error of the first two views

However, two of quantities in (17), (18) or (8) are only members of equivalence classes: due to the over-parametrization in formula (1). Formula (1) can be re-formulated as $\mathcal{H}_i \sim (\mathbf{R} + \mathbf{t}\Delta\mathbf{v}^T) + \mathbf{t}(\mathbf{v}_i^T - \Delta\mathbf{v}^T)$, i.e.,

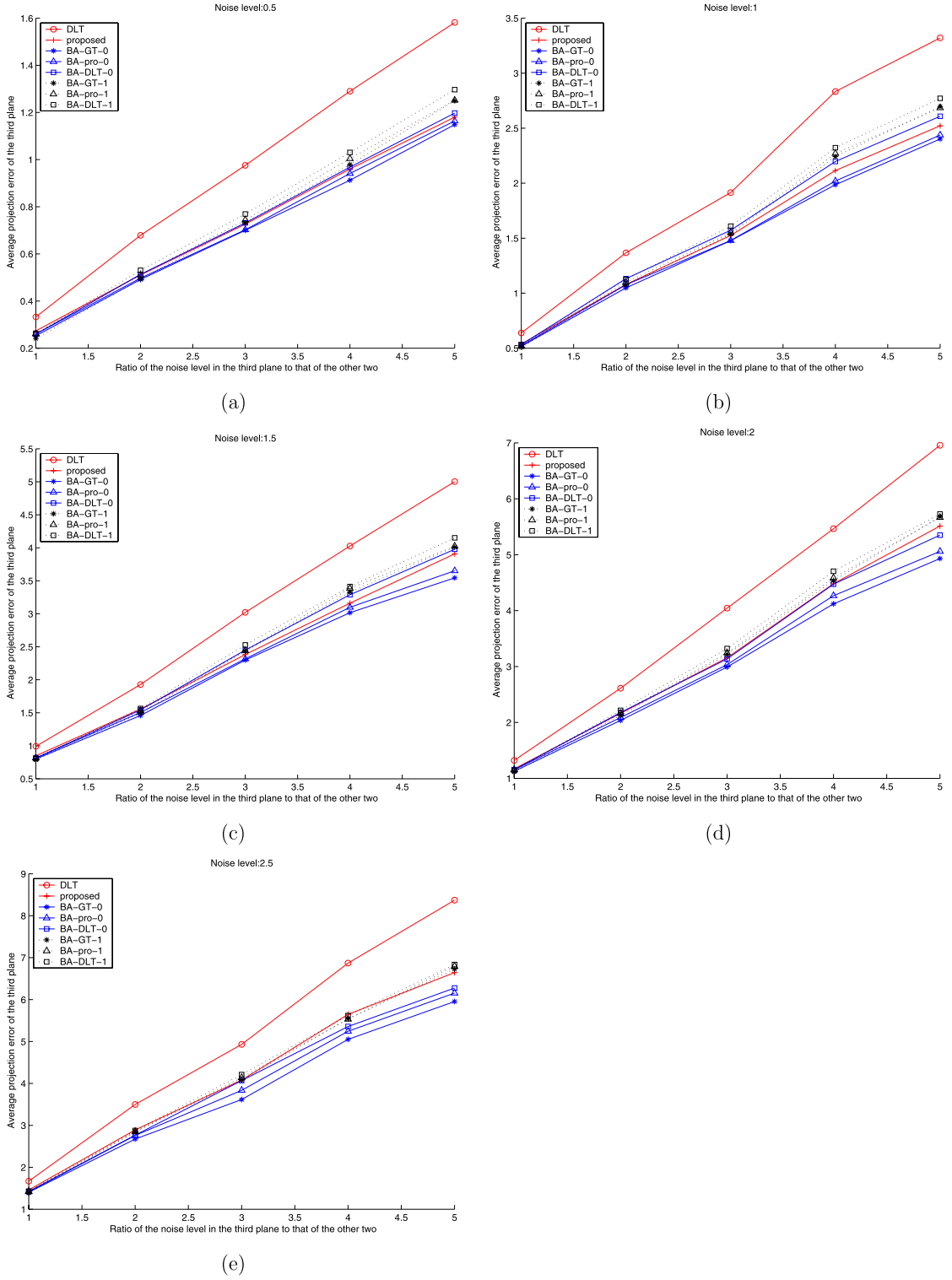
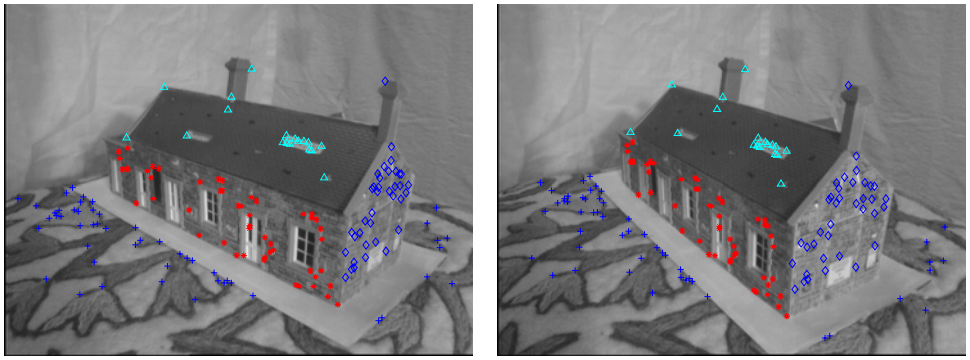


Fig. 3 Experimental result for 3-plane case: the third plane. In **a**–**e**, the abscissa is the ratio of the noise level of the third plane to that in the first two planes and the ordinate is the average error of the third view

$\mathbf{R} \leftarrow \mathbf{R} + \mathbf{t}\Delta\mathbf{v}^T$ and $\mathbf{v}_i^T \leftarrow \mathbf{v}_i^T - \Delta\mathbf{v}^T$ is another re-parameterization for a set of homographies. This over-parametrization results in an indefinite number of representations for a same set of homographies, in terms of $\mathbf{R}$,



The 8^{th} view The 9^{th} view

Fig. 4 Two views of the Model house

$\mathbf{t}$ and $\{\mathbf{v}_i\}$ in (1). Consequently, it is impossible to evaluate the estimated parameters, even if the ground truth data is available.

Regarding the parameter $\mathbf{t}$ (the translation vector) we can envisage an index for comparison: $r = \frac{\langle \tilde{\mathbf{t}}, \mathbf{t} \rangle}{\|\tilde{\mathbf{t}}\| \bullet \|\mathbf{t}\|}$, where $\tilde{\mathbf{t}}$ is the ground truth of the estimate $\mathbf{t}$. Figure 1 shows a plot of this quantity vs iteration count for a sample run on the Model House data. From Fig. 1, it can be seen the index r quickly reaches (and retains) a value of 1—indicating close agreement to the ground truth. However, this is not a practical measure (either for performance evaluation or as a stopping criterion as, of course, we do not know the ground truth in "real" applications.) Moreover, since it does not include any indication of the accuracy in terms of the other physically meaningful quantities, convergence in terms of this index does not necessarily equate to convergence to the ground truth result.

5.2 Validation without Knowing/Using Ground Truth

In this section we use image feature points direct from feature detectors: to demonstrate the effects of "realistic" noise distributions in the feature point locations—leading to realistic distributions of noise in the directly estimated homographies (without refinement). The "real" noise level will be influenced by many factors—the noise in the feature point locations, the spatial distribution of the feature points (more widely spread should lead to greater accuracy), and the number of feature points (the greater these are in number, generally will be the less noise); for example.

In this series of experiments, since we do not know the ground truth, we have to resort to demonstrating accuracy through image residuals from homography mapping error. That is, the difference between the second view and the projection of the first view on the second view is calculated (and scaled up to 2 times); and the darker the pixel, the bigger the difference. Note: the perceived accuracy is not only affected by the actual accuracy but also by such things as the texture (it is hard to see errors in regions of low texture and, conversely, in regions of high texture, small errors will look large). Not also that these experiments omit the "GT" variants as we now have no relevant ground truth for this setting.

Firstly, we use the Desktop images shown in Fig. 7. There are four planes: the computer screen, the table and two (right and left) calendars. Points are detected with the Harris corner detector and matched by hand. The four planes have 61, 17, 10 and 14 points, respectively. The back-projections of homographies are shown in Figs. 8–11: DLT, proposed, BA-pro-0, BA-DLT-0, BA-pro-1 and BA-DLT-1. As a second series of experiments using the model house data, we estimate the homographies, using real image feature point data (not our "purified" points with synthetic noise used in Sect. 5.1.2). The back-projection of the first view on the second view is shown in Figs. 12–15.

The main conclusion is that DLT performs worse, as expected, than any of the refinement methods. This is particularly evident in all planes from the Desktop images (except the computer screen) and in the house-side plane (Fig. 15). For the computer screen, there are a significant number of matched points (indeed, for the other planes in the Desktop scene there are so few matches that it is not surprising that the DLT estimate is poor). In other words, by refinement, one can generally "get away" with using less points on one or more of the planes, as the inaccuracy will be reduced by the enforcement of coherence amongst all homography estimates.

Visual inspection of pixel residuals does not always reveal small discrepancies: it is hard to distinguish the quality improvement amongst the refinement methods (ours and the variants of bundle adjustment) but clearly our method is competitive. Nonetheless, we have demonstrated, by the visual improvements on most planes, that the refinement methods, ours and bundle adjustment, can improve homography estimation even in situations using a few planes for the refinement.

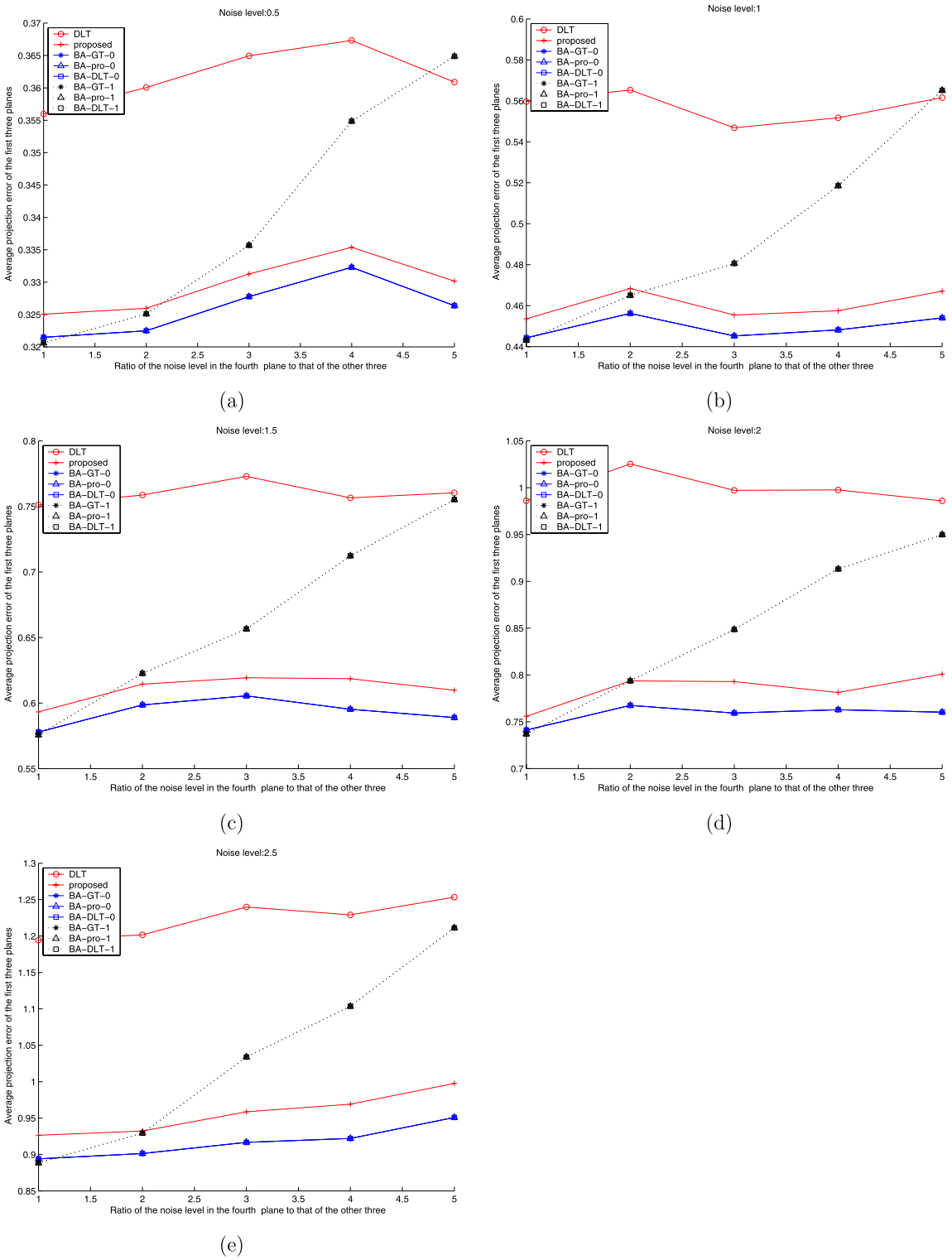


Fig. 5 Experimental result for the house model: the three planes with equal noise added (roof, ground, front). In **b–f**, the abscissa is the ratio of the noise level of the fourth plane to that in the first three planes and the ordinate is the average error of the first three views

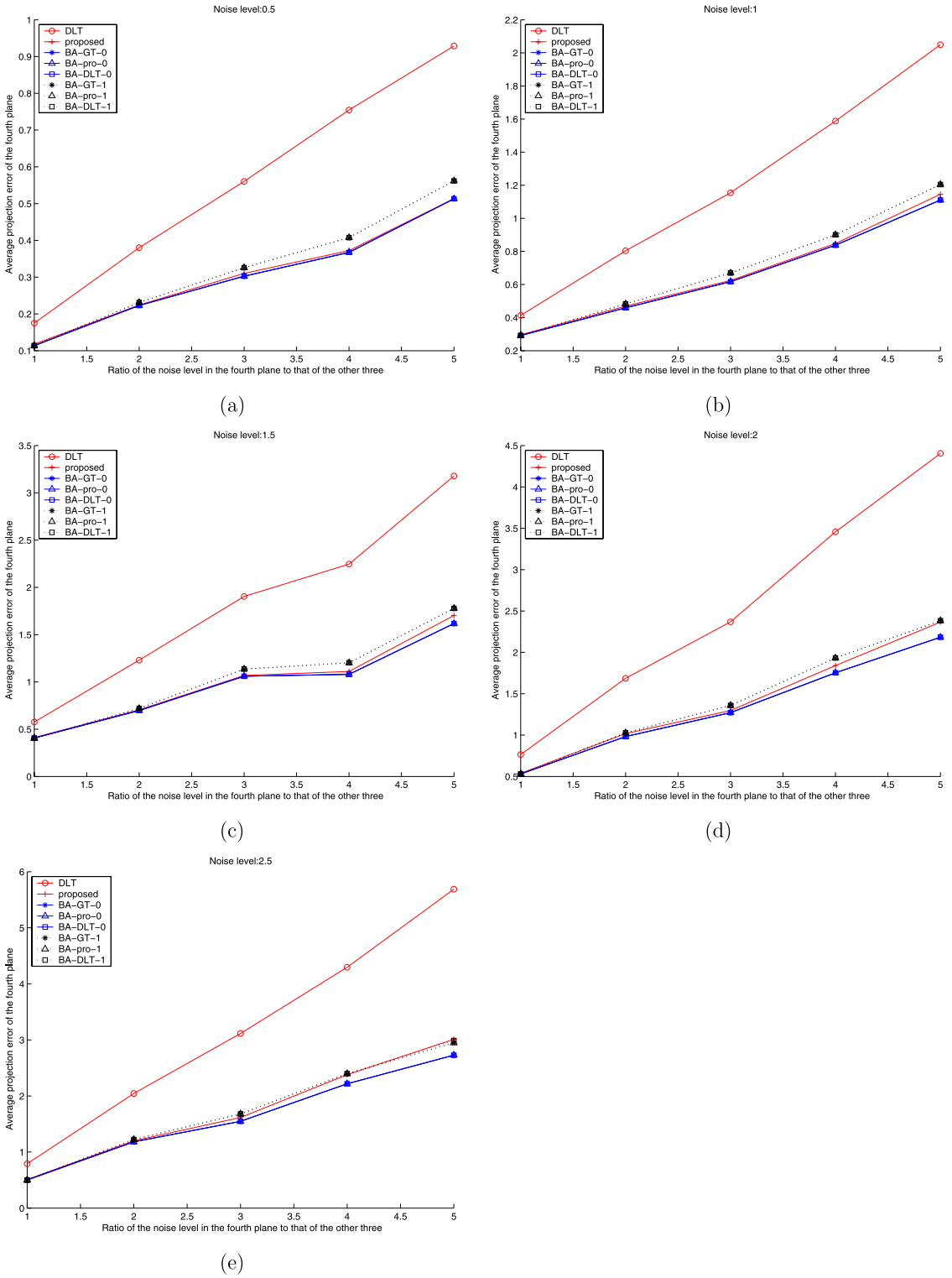


Fig. 6 Experimental result for the house model: the plane with more added noise (side of the house). In **a**–**e**, the abscissa is the ratio of the noise level of the fourth plane to that in the first three planes and the ordinate is the average error of the fourth view

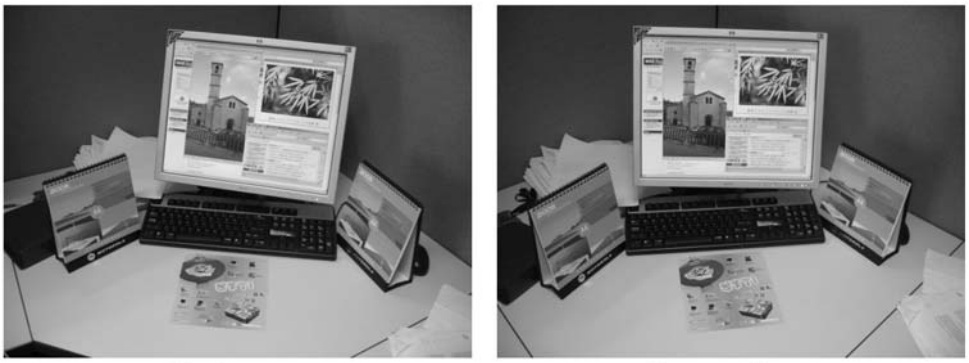


Fig. 7 Two views of the desktop

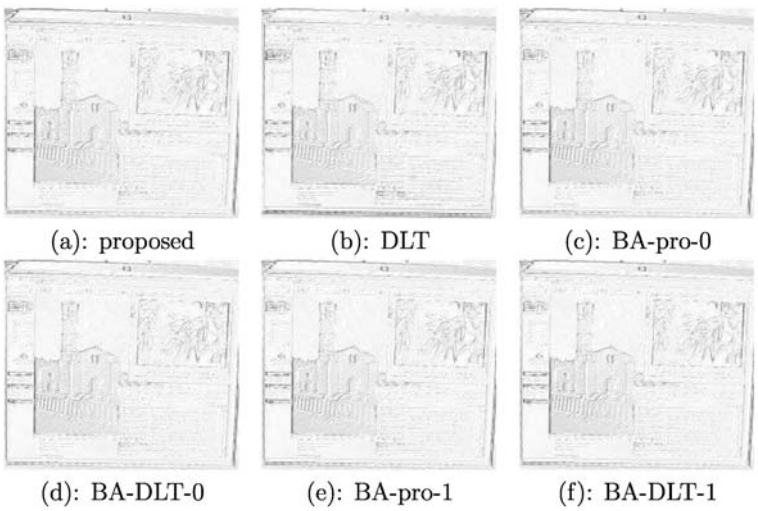


Fig. 8 Experimental result for the plane of computer screen

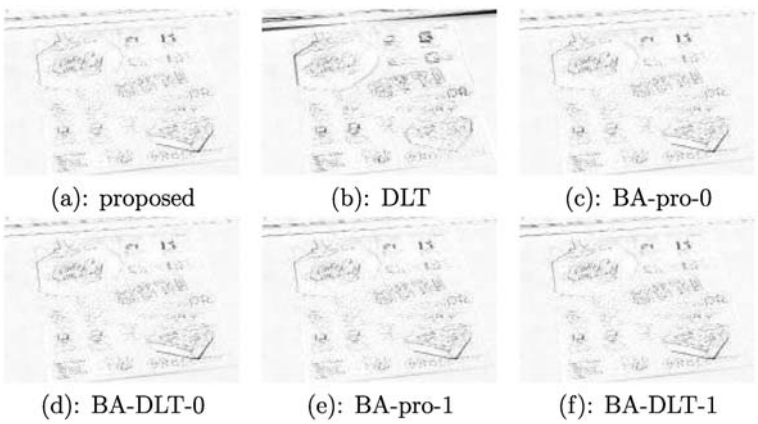


Fig. 9 Experimental result for the plane on the table

5.3 Complexity Analysis

Recall (Golub and Loan 1996) that the complexity of solving a linear equation $\mathbf{Ax} = \mathbf{b}$, with $\mathbf{A} \in R^{m,n}$ $(m \geq n)$, is of $O(mn^2)$.

The main computational burden of the proposed algorithm lies in the core algorithm. Taking into account of the fact that the noise in each homography is independent of that in another homography, computing $\mathbf{t}$, $\mathbf{v}$, $\mathbf{u}$ and $\mathbf{d}$ in (9) involves solving $9n \times 3$ linear equations, n 9×3 equations, a $9n \times 9$ and n 9×1 equations, respectively; and their complexity is $O(9n \times 3^2)$, $O(n \times 9 \times 3^2)$, $O(9n \times 9^2)$ and $O(n \times 9)$, respectively. Thus the overall complexity is $O(n)$.

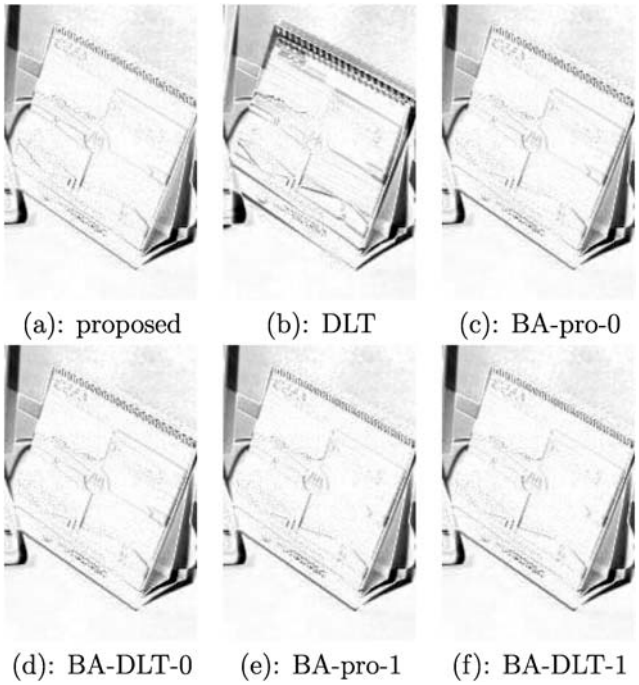


Fig. 10 Experimental result for the plane of the right calendar

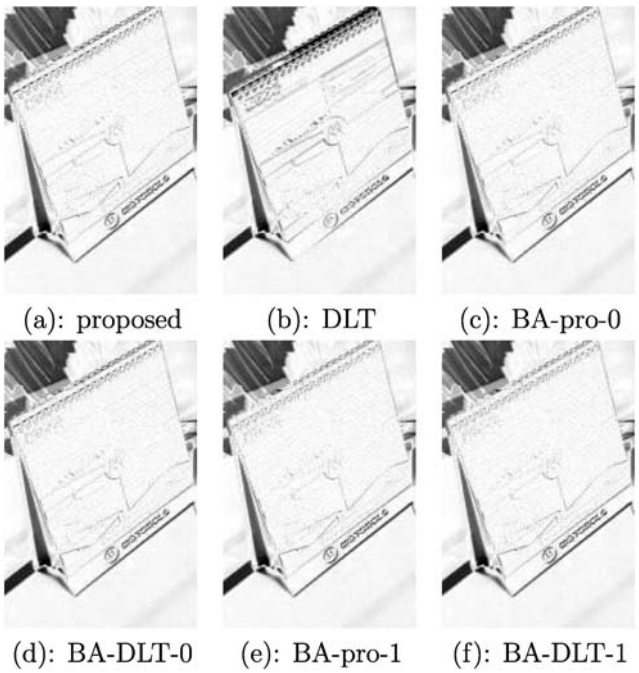


Fig. 11 Experimental result for the plane of the left calendar

One iteration of the BA algorithm involves solving a $(12+3n)\times(12+3n)$ linear equation and constructing the $(12+3n)\times(12+3n)$ Hessian matrix. The solution of the $(12+3n)\times(12+3n)$ linear equation has a complexity of $O((3n+12)^3)$, and the construction of the Hessian matrix has a complexity of $O(\sum_i p_i(3n+12)^2)$, where p_i is the number of feature points in the ith plane. The complexity of using the DLT algorithm to compute the ith homography is $O(2p_i 9^2)$, and for n homographies, is $O(2\sum_i p_i 9^2)$.

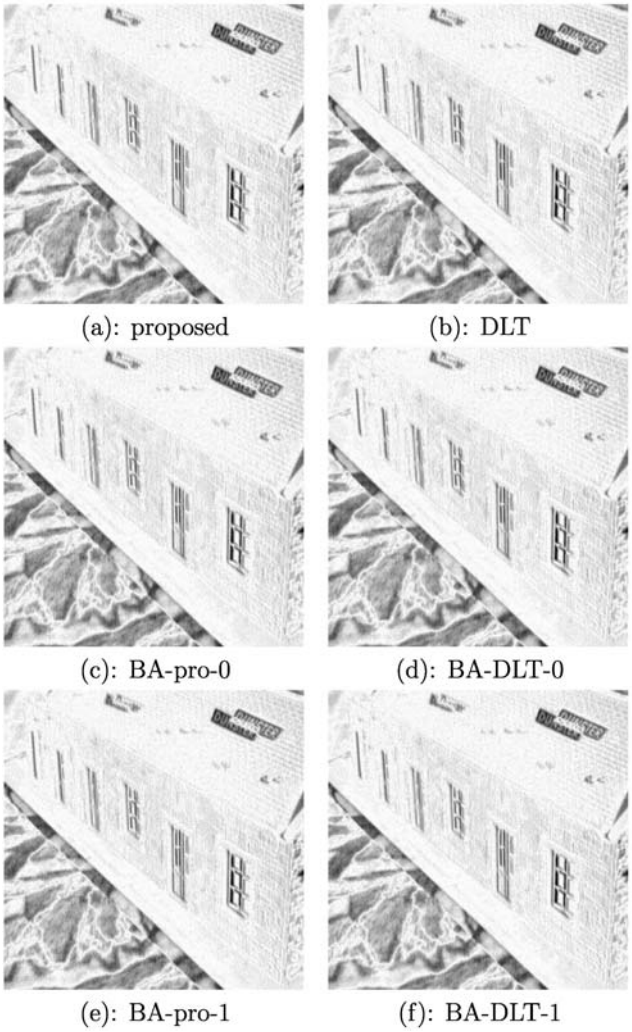


Fig. 12 Experimental result for the plane of the front wall. The reader should ignore all parts of the image except the region of the front wall

Obviously, the DLT algorithm runs much faster than the other two.

This complexity analysis would suggest that our method should outperform the BA algorithm, in terms of speed, for large n. However, it would be disingenuous to put stress on this as a virtue of our scheme, as we have emphasized the practical need to address a modest number of homographies.

In terms of empirical running speed, in our code, our method is about twice as slow as BA. However, we have made no attempt to optimize our code and the minimization code of BA is, presumably, from a highly optimized library. All we can conclude is that our algorithm is roughly competitive in speed (unoptimized) and may be made more competitive if effort is put into optimization.

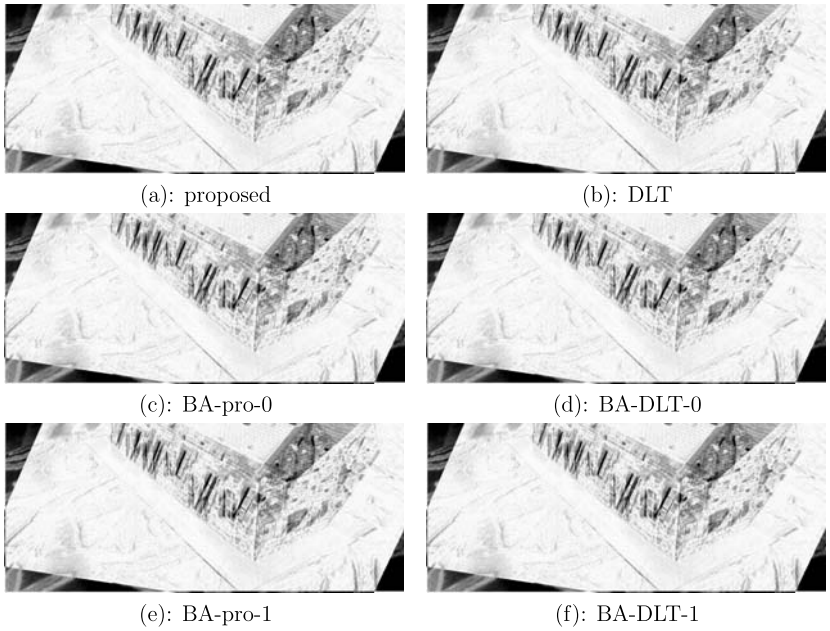


Fig. 13 Experimental result for the plane of the ground floor. The reader should ignore all parts of the image except the region of the ground

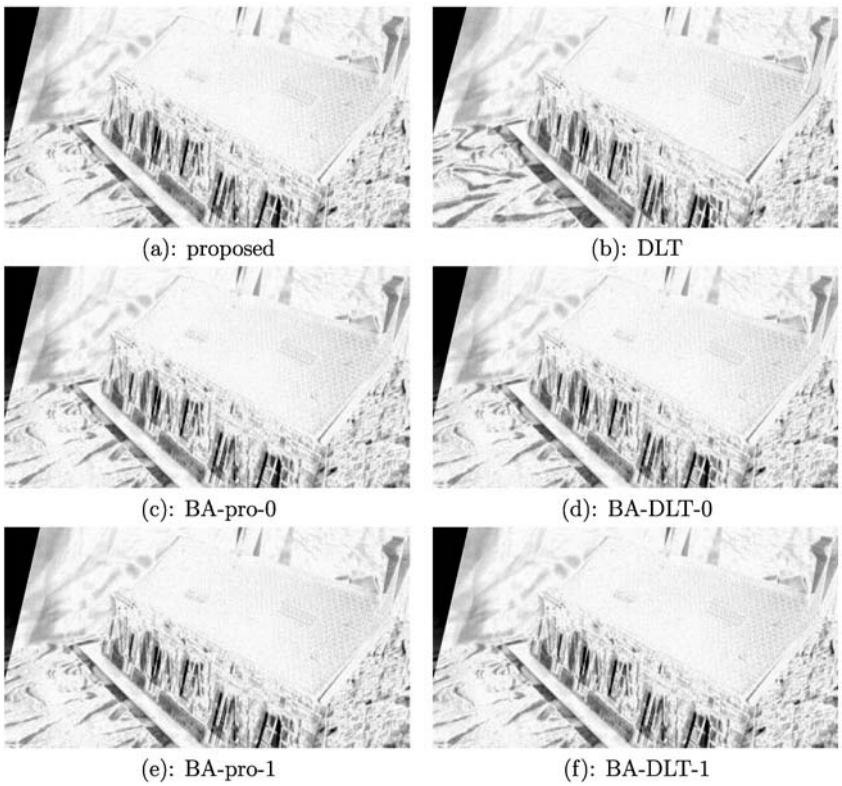


Fig. 14 Experimental result for the roof plane. The reader should ignore all parts of the image except the roof

6 Conclusion

In this paper, we study how to jointly estimate ≥ 3 homographies over two views. There are three major contributions. First, the dimension-three affine homography subspace can be calculated from ≥ 3 homographies, which is contrary to the naive view that four homographies are needed to do this; more importantly, the homography matrix can be decomposed into the "combination" of two rank-one matrices so that more constraints are available to reliably improve homography accuracy, in cases with a realistic number of planes. Second, we show how to estimate the subspace constrained homography. Third, we show how to statistically improve the joint estimation of several homographies.

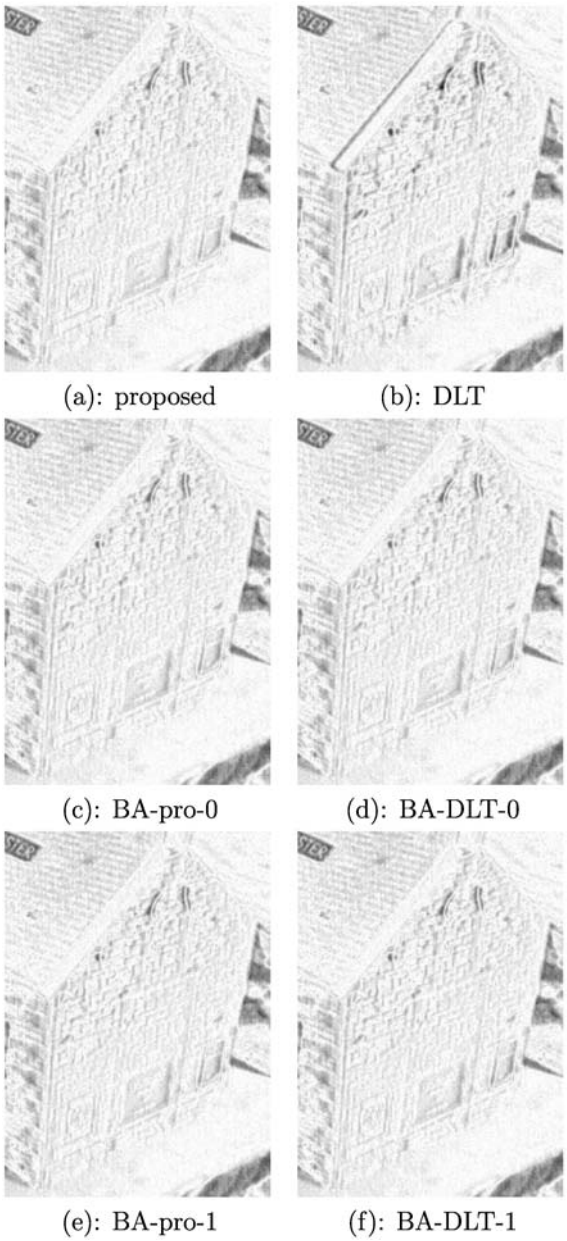


Fig. 15 Experimental result for the plane of the side wall. The reader should ignore all parts of the image except the side wall

Combining these principles, can produce an algorithm that greatly improves over separately estimating the homographies. Indeed, compared with the "gold standard" of bundle adjustment (when starting from a "good" starting point), our experiments show that our method is close to the result of that "gold standard". Our method (like bundle adjustment) is iterative but provably convergent (albeit to a local minimum). Experimentally, we demonstrate that it generally converges to a very good (perhaps global) minimum with at most two starting points (particular choices proposed as part of our strategy).

We should emphasize that we are not claiming that we have produced an algorithm that unequivocally outperforms the current "state of the art" (BA initialized from DLT). Indeed, compared with that, our accuracy gains can be modest (though reliably better on average). Such gains come, currently, by at least as much computational cost (which may be improved, however, by careful crafting of the implementation).

One might even consider a practical algorithm using BA initialized by our approach. Indeed, a particularly useful scenario may be to use our algorithm, test the accuracy and if still not accurate enough, attempt to further refine through BA. Moreover, we have shown the end result is generally better than BA initialized by DLT and very close to the best one can hope for by any variant of initializing BA (as shown by comparing against that initialized through the ground truth data).

The contributions of this paper extend beyond that of proposing an algorithm. We have shown why the naive approach of direct rank-constraint is doomed to produce modest improvement. We have shown how there are further rank constraints to take advantage of. Thus we have added to the understanding of rank-constraints in the homography estimation setting.

Acknowledgement We thank the Robotics Research Group, University of Oxford, for providing the data of the Model House. We would like to thank the anonymous reviewers for their insightful comments and questions, which greatly improved the manuscript. Pei Chen thanks the support from the Ministry of Science and Technology, The People's Republic of China (International Science and Technology Cooperation Projects 2006DFB73360).

Appendix A: Definition of the Matrix $\mathbf{M}_{i,j}$ in (4)

Suppose two homographies with matrix representations $\mathcal{H}_i$ and $\mathcal{H}_j$ are estimated, up to different unknown scales (all that can be done in practice): $\mathcal{H}_i \sim (\mathbf{R} + \mathbf{t}\mathbf{v}_i^T)$ and $\mathcal{H}_j \sim (\mathbf{R} + \mathbf{t}\mathbf{v}_j^T)$.

Note: there exist two independent 3-vectors $\mathbf{l}_i = [l_{i,x}\ l_{i,y}\ l_{i,z}]^T$, for $i = 1, 2$, that span $\mathbf{t}^\perp$ (i.e., $span\{\mathbf{l}_1, \mathbf{l}_2\} = \mathbf{t}^\perp$), where $\mathbf{t}^\perp$ denotes the dimension-two subspace that is perpendicular to the vector $\mathbf{t}$, and further

$$\mathbf{l}^T\mathcal{H}_i \sim \mathbf{l}^T\mathcal{H}_j \tag{21}$$

where $\mathbf{l} = [l_x\ l_y\ l_z]^T = c_1\mathbf{l}_1 + c_2\mathbf{l}_2 \in span\{\mathbf{l}_1, \mathbf{l}_2\}$.

From (21), their cross product is a zero vector, i.e., $[\mathcal{H}_i^T\mathbf{l}]_\times\mathcal{H}_j^T\mathbf{l} = \mathbf{0}$, for any c_1 and c_2. In column vector notation, we write the cross product as:

$$\mathbf{M}_{i,j}\mathbf{L} = \mathbf{0} \tag{22}$$

where $[\mathbf{L}]_{6\times 1} = [l_x^2 \;\; l_y^2 \;\; l_z^2 \;\; l_x l_y \;\; l_x l_z \;\; l_y l_z]^T$ and $\mathbf{M}_{i,j}$ (size 3×6) is defined below.

$$\mathbf{M}_{i,j} = \begin{bmatrix} h_{1,i}h_{2,j} - h_{1,j}h_{2,i} & h_{4,i}h_{5,j} - h_{4,j}h_{5,i} & & & & \\ h_{7,i}h_{8,j} - h_{7,j}h_{8,i} & M_{i,j}^{1,4} & M_{i,j}^{1,5} & M_{i,j}^{1,6} & & \\ h_{1,i}h_{3,j} - h_{1,j}h_{3,i} & h_{4,i}h_{6,j} - h_{4,j}h_{6,i} & & & & \\ h_{7,i}h_{9,j} - h_{7,j}h_{9,i} & M_{i,j}^{2,4} & M_{i,j}^{2,5} & M_{i,j}^{2,6} & & \\ h_{2,i}h_{3,j} - h_{2,j}h_{3,i} & h_{5,i}h_{6,j} - h_{5,j}h_{6,i} & & & & \\ h_{8,i}h_{9,j} - h_{8,j}h_{9,i} & M_{i,j}^{3,4} & M_{i,j}^{3,5} & M_{i,j}^{3,6} & & \end{bmatrix}$$

with

$$M_{i,j}^{1,4} = h_{1,i}h_{5,j} - h_{1,j}h_{5,i} + h_{4,i}h_{2,j} - h_{4,j}h_{2,i},$$
$$M_{i,j}^{1,5} = h_{1,i}h_{8,j} - h_{1,j}h_{8,i} + h_{7,i}h_{2,j} - h_{7,j}h_{2,i},$$
$$M_{i,j}^{1,6} = h_{4,i}h_{8,j} - h_{4,j}h_{8,i} + h_{7,i}h_{5,j} - h_{7,j}h_{5,i},$$
$$M_{i,j}^{2,4} = h_{1,i}h_{6,j} - h_{1,j}h_{6,i} + h_{4,i}h_{3,j} - h_{4,j}h_{3,i},$$
$$M_{i,j}^{2,5} = h_{1,i}h_{9,j} - h_{1,j}h_{9,i} + h_{7,i}h_{3,j} - h_{7,j}h_{3,i},$$
$$M_{i,j}^{2,6} = h_{4,i}h_{9,j} - h_{4,j}h_{9,i} + h_{7,i}h_{6,j} - h_{7,j}h_{6,i},$$
$$M_{i,j}^{3,4} = h_{2,i}h_{6,j} - h_{2,j}h_{6,i} + h_{5,i}h_{3,j} - h_{5,j}h_{3,i},$$
$$M_{i,j}^{3,5} = h_{2,i}h_{9,j} - h_{2,j}h_{9,i} + h_{8,i}h_{3,j} - h_{8,j}h_{3,i},$$

and

$$M_{i,j}^{3,6} = h_{5,i}h_{9,j} - h_{5,j}h_{9,i} + h_{8,i}h_{6,j} - h_{8,j}h_{6,i}.$$

Appendix B: Proof that Matrix M in (4) Has a Rank of Three

Define

$$[\mathbf{L}_1]_{6\times 1} = [l_{1,x}^2 \;\; l_{1,y}^2 \;\; l_{1,z}^2 \;\; l_{1,x}l_{1,y} \;\; l_{1,x}l_{1,z} \;\; l_{1,y}l_{1,z}]^T,$$
$$[\mathbf{L}_2]_{6\times 1} = [l_{2,x}^2 \;\; l_{2,y}^2 \;\; l_{2,z}^2 \;\; l_{2,x}l_{2,y} \;\; l_{2,x}l_{2,z} \;\; l_{2,y}l_{2,z}]^T$$

and

$$[\mathbf{L}_{1,2}]_{6\times 1} = [2l_{1,x}l_{2,x} \;\; 2l_{1,y}l_{2,y} \;\; 2l_{1,z}l_{2,z} \;\; l_{1,x}l_{2,y} + l_{2,x}l_{1,y} \;\; l_{1,x}l_{2,z} + l_{2,x}l_{1,z} \;\; l_{1,y}l_{2,z} + l_{2,y}l_{1,z}]^T.$$

As special cases of (22), we note that: $\mathbf{M}_{i,j}\mathbf{L}_1 = \mathbf{0}$ and $\mathbf{M}_{i,j}\mathbf{L}_2 = \mathbf{0}$. Since $\mathbf{L} = c_1^2\mathbf{L}_1 + c_2^2\mathbf{L}_2 + c_1c_2\mathbf{L}_{1,2}$ and $\mathbf{M}_{i,j}\mathbf{L} = \mathbf{0}$ holds for any c_1, c_2, $\mathbf{M}_{i,j}\mathbf{L}_{1,2} = \mathbf{0}$ also holds.

Obviously, $\mathbf{L}_1$ and $\mathbf{L}_2$ are independent. In addition, from the definition of $\mathbf{L}$ (immediately following (22)), $\mathbf{L} = \mathbf{0}$ *iff* $l_x = l_y = l_z = 0$, i.e., $\mathbf{l} = \mathbf{0}$. Moreover, because $\mathbf{l}_1$ and $\mathbf{l}_2$ are independent, $\mathbf{l} = \mathbf{0}$ *iff* $c_1 = c_2 = 0$. Thus, $\mathbf{L}_1$, $\mathbf{L}_2$ and $\mathbf{L}_{1,2}$ are independent vectors.

Combining these facts, there exists a dimension-three subspace: $span\{\mathbf{L}_1, \mathbf{L}_2, \mathbf{L}_{1,2}\}$, which is contained in the null space of $\mathbf{M}_{i,j}$ for $i < j$. $span\{\mathbf{L}_1, \mathbf{L}_2, \mathbf{L}_{1,2}\}$ is contained in the null space of $\mathbf{M}$, because the subspace of $span\{\mathbf{L}_1, \mathbf{L}_2, \mathbf{L}_{1,2}\}$ is independent of c_1 and c_2.

Thus, $rank(\mathbf{M}) \leq 6 - 3$, because $\mathbf{M}$ has a width of six. Assuming non-degeneracy (the planes are not parallel) $rank(\mathbf{M}) = 3$.

Appendix C: Proof that t is the Left Null Vector of S in (5)

In practice, we calculate the dimension-three null space of $\mathbf{M}$ up to a nonsingular 3×3 transform $\mathbf{Y}$. That is, the basis vectors spanning the null-space are: $[\mathbf{N}]_{6\times 3} = [\mathbf{n}_1 \; \mathbf{n}_2 \; \mathbf{n}_3] = [\mathbf{L}_1 \; \mathbf{L}_2 \; \mathbf{L}_{1,2}]_{6\times 3}[\mathbf{Y}]_{3\times 3}$. Each column of $\mathbf{N}$ is arranged as a 3×3 matrix $\mathcal{N}_i$, as defined in (6).

Now, we prove that $\mathbf{t}^T\mathcal{N}_i = \mathbf{0}$.

If $\mathbf{Y}$ is an identity matrix, i.e., $\mathbf{Y} = \mathbf{I}_3$, $\mathcal{N}_i' = \mathbf{l}_i^T\mathbf{l}_i$ for $(i = 1, 2)$, and $\mathcal{N}_3' = \mathbf{l}_1^T\mathbf{l}_2 + \mathbf{l}_2^T\mathbf{l}_1$. Then $\mathbf{t}^T\mathcal{N}_i' = \mathbf{0}$ holds for $i = 1, 2, 3$, because of $\mathbf{l}_i \in \mathbf{t}^\perp$ for $i = 1, 2$. For any nonsingular $\mathbf{Y}$, $\mathcal{N}_i = \sum_{k=1}^3 y_{k,i}\mathcal{N}_k'$, because the column of $\mathbf{N}$ is a linear combination of $\mathbf{L}_1$, $\mathbf{L}_1$ and $\mathbf{L}_{1,2}$. Thus $\mathbf{t}^T\mathcal{N}_i = \mathbf{0}$ also holds.

In order to make full use of the available data, we juxtapose $\mathcal{N}_i$ as:

$$[\mathbf{S}]_{3\times 9} = [\mathcal{N}_1 \;\; \mathcal{N}_2 \;\; \mathcal{N}_3]_{3\times 9} = [\mathbf{l}_1\mathbf{l}_1^T \;\; \mathbf{l}_2\mathbf{l}_2^T \;\; \mathbf{l}_1\mathbf{l}_2^T + \mathbf{l}_2\mathbf{l}_1^T]_{3\times 9}[(\mathbf{Y} \otimes \mathbf{I}_3)]_{9\times 9} \quad (23)$$

Because $rank(\mathbf{A} \otimes \mathbf{B}) = rank(\mathbf{A})rank(\mathbf{B})$, $\mathbf{Y} \otimes \mathbf{I}_3$ has a full rank of nine. $[\mathbf{l}_1\mathbf{l}_1^T \;\; \mathbf{l}_2\mathbf{l}_2^T]$ has a rank of two because $\mathbf{l}_1$ and $\mathbf{l}_2$ are independent, and $\mathbf{t}$ is its unique left null vector. So $rank(\mathbf{S}) \geq 2$. Also, $\mathbf{t}^T\mathbf{S} = \mathbf{0}$ and consequently $rank(\mathbf{S}) \leq 2$. Thus, $\mathbf{S}$ has a rank of two and the solution of $\mathbf{t}$, up to an unknown scale, is the left null vector of $\mathbf{S}$.

Appendix D: Error in Homography Coefficients

In this appendix, we summarize the results in Chen and Suter (2008), showing how to compute the covariance matrix of the error in homography coefficients. The aim of the statistical analysis in Chen and Suter (2008) is to represent the error in the homography parameters in terms of the random variables of $\{\varepsilon_{i,1}, \varepsilon_{i,2}, \varepsilon_{i,1}', \varepsilon_{i,2}'\}$ for $(1 \leq i \leq n)$, which are the errors in the coordinates of 2D feature matches:

$$\boldsymbol{\eta}_k(\mathbf{h}_k) = \Xi_{\mathbf{h}_k}\mathbf{e} \quad (24)$$

where $[\mathbf{e}]_{4n\times 1} = [\varepsilon_{1,1} \; \varepsilon_{1,2} \; \varepsilon_{1,1}' \; \varepsilon_{1,2}' \; \ldots \; \varepsilon_{n,1} \; \varepsilon_{n,2} \; \varepsilon_{n,1}' \; \varepsilon_{n,2}']^T$. $\boldsymbol{\eta}(\bullet)$ here is used to denote the error in the quantity $\bullet$.

From (24), we can then arrive at our required covariance matrix

$$\mathbf{C}_{\mathbf{h}_k} = \Xi_{\mathbf{h}_k} \Pi \Xi_{\mathbf{h}_k}^T \tag{25}$$

where Π is the $4n \times 4n$ covariance matrix for the noise $\mathbf{e}$ in the image points. In the special case, where i.i.d. 0-mean-σ^2-variance Gaussian (feature point) noise is assumed, the error covariance matrix in the homography is

$$\mathbf{C}_{\mathbf{h}_k} = \sigma^2 \Xi_{\mathbf{h}_k} \Xi_{\mathbf{h}_k}^T \tag{26}$$

Furthermore, the noise level σ can be accurately estimated by employing first-order approximation techniques (Chen and Suter 2008).

Appendix E: Error in the Matrix M in (4)

The error in the matrix $\mathbf{M}$ in (4) can not be considered as i.i.d. Gaussian, due to two reasons: the heteroscedastic error in the homography parameters and the non-linearity introduced in the calculation of $\mathbf{M}$.

Writing the error in each homography $\mathbf{h}_i$ as $\eta_i = [\eta_{i,1} \ \dots \ \eta_{i,9}]^T$, and collecting to form the error in the n homographies: $\eta = [\eta_1^T \ \dots \ \eta_n^T]^T$. The errors in each homography are assumed to be of zero mean and independent from those in another homography.

Unlike the homography matrix, the errors in the matrix $\mathbf{M}$ can not be considered to be column independent (thus they are both row and column dependent). We arrange the $3C_n^2 \times 6$ matrix $\mathbf{M}$ as a $18C_n^2 \times 1$ vector: $vec(\mathbf{M}^T)$. The error in this vector, denoted by ζ, is related to the error in the homographies by:

$$\zeta = \Phi \eta \tag{27}$$

where Φ is an $18C_n^2 \times 9n$ matrix,

$$\Phi = \left[\Phi_{1,2}^T \ \dots \ \Phi_{1,n}^T \ \Phi_{2,3}^T \ \dots \ \Phi_{2,n}^T \ \dots \ \Phi_{n-1,n}^T\right]^T \tag{28}$$

and with the $18 \times 9n$ matrix: $\Phi_{i,j}$ given at the end of this appendix.

From (27), the covariance matrix of ζ is

$$\mathbf{C}_\zeta = \Phi E(\eta\eta^T)\Phi^T \tag{29}$$

where $E(\eta\eta^T)$ is a block diagonal matrix: $E(\eta\eta^T) = diag\{E(\eta_i\eta_i^T)\}$ because η_i is independent of η_j for $i \neq j$.

The covariance matrix $E(\eta_i\eta_i^T)$ can be calculated from the feature points while calculating the homography—see (25) and (26).

$$\Phi_{i,j} = \begin{bmatrix} \mathbf{0}_{6,9(i-1)} & \phi_{i,j,1,2} & \mathbf{0}_{6,9(j-i-1)} & \phi_{i,j,1,4} & \mathbf{0}_{6,9(n-j)} \\ \mathbf{0}_{6,9(i-1)} & \phi_{i,j,2,2} & \mathbf{0}_{6,9(j-i-1)} & \phi_{i,j,2,4} & \mathbf{0}_{6,9(n-j)} \\ \mathbf{0}_{6,9(i-1)} & \phi_{i,j,3,2} & \mathbf{0}_{6,9(j-i-1)} & \phi_{i,j,3,4} & \mathbf{0}_{6,9(n-j)} \end{bmatrix}$$

where

$$\phi_{i,j,1,2} = \begin{bmatrix} h_{2,j} & -h_{1,j} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_{5,j} & -h_{4,j} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_{8,j} & -h_{7,j} & 0 \\ h_{5,j} & -h_{4,j} & 0 & h_{2,j} & -h_{1,j} & 0 & 0 & 0 & 0 \\ h_{8,j} & -h_{7,j} & 0 & 0 & 0 & 0 & h_{2,j} & -h_{1,j} & 0 \\ 0 & 0 & 0 & h_{8,j} & -h_{7,j} & 0 & h_{5,j} & -h_{4,j} & 0 \end{bmatrix}$$

$$\phi_{i,j,1,4} = \begin{bmatrix} -h_{2,i} & h_{1,i} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -h_{5,i} & h_{4,i} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -h_{8,i} & h_{7,i} & 0 \\ -h_{5,i} & h_{4,i} & 0 & -h_{2,i} & h_{1,i} & 0 & 0 & 0 & 0 \\ -h_{8,i} & h_{7,i} & 0 & 0 & 0 & 0 & -h_{2,i} & h_{1,i} & 0 \\ 0 & 0 & 0 & -h_{8,i} & h_{7,i} & 0 & -h_{5,i} & h_{4,i} & 0 \end{bmatrix}$$

$$\phi_{i,j,2,2} = \begin{bmatrix} h_{3,j} & 0 & -h_{1,j} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_{6,j} & 0 & -h_{4,j} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_{9,j} & 0 & -h_{7,j} \\ h_{6,j} & 0 & -h_{4,j} & h_{3,j} & 0 & -h_{1,j} & 0 & 0 & 0 \\ h_{9,j} & 0 & -h_{7,j} & 0 & 0 & 0 & h_{3,j} & 0 & -h_{1,j} \\ 0 & 0 & 0 & h_{9,j} & 0 & -h_{7,j} & h_{6,j} & 0 & -h_{4,j} \end{bmatrix}$$

$$\phi_{i,j,2,4} = \begin{bmatrix} -h_{3,i} & 0 & h_{1,i} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -h_{6,i} & 0 & h_{4,i} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -h_{9,i} & 0 & h_{7,i} \\ -h_{6,i} & 0 & h_{4,i} & -h_{3,i} & 0 & h_{1,i} & 0 & 0 & 0 \\ -h_{9,i} & 0 & h_{7,i} & 0 & 0 & 0 & -h_{3,i} & 0 & h_{1,i} \\ 0 & 0 & 0 & -h_{9,i} & 0 & h_{7,i} & -h_{6,i} & 0 & h_{4,i} \end{bmatrix}$$

$$\phi_{i,j,3,2} = \begin{bmatrix} 0 & h_{3,j} & -h_{2,j} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_{6,j} & -h_{5,j} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_{9,j} & -h_{8,j} \\ 0 & h_{6,j} & -h_{5,j} & 0 & h_{3,j} & -h_{2,j} & 0 & 0 & 0 \\ 0 & h_{9,j} & -h_{8,j} & 0 & 0 & 0 & 0 & h_{3,j} & -h_{2,j} \\ 0 & 0 & 0 & 0 & h_{9,j} & -h_{8,j} & 0 & h_{6,j} & -h_{5,j} \end{bmatrix}$$

$$\phi_{i,j,3,4} = \begin{bmatrix} 0 & -h_{3,i} & h_{2,i} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -h_{6,i} & h_{5,i} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -h_{9,i} & h_{8,i} \\ 0 & -h_{6,i} & h_{5,i} & 0 & -h_{3,i} & h_{2,i} & 0 & 0 & 0 \\ 0 & -h_{9,i} & h_{8,i} & 0 & 0 & 0 & 0 & -h_{3,i} & h_{2,i} \\ 0 & 0 & 0 & 0 & -h_{9,i} & h_{8,i} & 0 & -h_{6,i} & h_{5,i} \end{bmatrix}$$

References

Aguiar, P. M. Q., & Moura, J. M. F. (1999a). Factorization as a rank 1 problem. In *Proc. conf. computer vision and pattern recognition* (pp. 178–184).

Aguiar, P. M. Q., & Moura, J. M. F. (1999b). A fast algorithm for rigid structure from image sequences. In *Proc. int'l conf. image processing*.

Aguiar, P. M. Q., & Moura, J. M. F. (2000). Weighted factorization. In *ICIP* (pp. 549–552).

Aguiar, P. M. Q., & Moura, J. M. F. (2001). Three-dimensional modeling from two-dimensional video. *IEEE Transactions on Image Processing*, *10*(10).

Aguiar, P. M. Q., & Moura, J. M. F. (2003). Rank 1 weighted factorization for 3d structure recovery: algorithms and performance analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(9), 1134–1149.

Anandan, P., & Irani, M. (2002). Factorization with uncertainty. *International Journal of Computer Vision*, *49*(2/3), 101–116.

Basri, R., & Jacobs, D. W. (1999). Lambertian reflectance and linear subspaces. In *Proc. int'l conf. computer vision* (pp. 383–390).

Basri, R., & Jacobs, D. W. (2003). Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(2), 218–233.

Brand, M. (2002). Incremental singular value decomposition of uncertain data with missing values. In *Proc. European conf. computer vision* (pp. 707–720).

Chen, P. (2004). *An investigation of statistical aspects of linear subspace analysis for computer vision applications*. PhD thesis, Monash University.

Chen, P., & Suter, D. (2008, to appear). Error analysis in homography estimation by first order approximation tools: A general technique. *Journal of Mathematical Imaging and Vision*.

Chen, P., & Suter, D. (2006). An analysis of linear subspace approaches for computer vision and pattern recognition. *International Journal of Computer Vision*, *68*(1), 83–106.

Chen, P., & Suter, D. (2007). A bilinear approach to the parameter estimation of a general heteroscedastic linear system, with application to conic fitting. *Journal of Mathematical Imaging and Vision*, *28*(3), 191–208.

Fischler, M. A., & Rolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, *24*(6), 381–395.

Golub, G. H., & Loan, C. F. V. (1996). *Matrix computations* (3nd edn.). Baltimore: Johns Hopkins University Press.

Hartley, R. I., & Zisserman, A. (2003). *Multiple view geometry in computer vision* (2nd edn.). Cambridge: Cambridge University Press.

Irani, M., & Anandan, P. (2000). Factorization with uncertainty. In *Proc. European conf. computer vision* (Vol. 49, pp. 539–553).

Leedan, Y., & Meer, P. (2000). Heteroscedastic regression in computer vision: Problems with bilinear constraint. *International Journal of Computer Vision*, *37*(2), 127–150.

Manton, J. H., Mahony, R., & Hua, Y. (2003). The geometry of weighted low-rank approximations. *IEEE Transactions on Signal Processing*, *51*(2), 500–514.

Morita, T., & Kanade, T. (1997). A sequential factorization method for recovering shape and motion from image streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(8), 858–867.

Poelman, C., & Kanade, T. (1994). A paraperspective factorization method for shape and motion recovery. In *Proc. European conf. computer vision* (pp. 206–218).

Poelman, C., & Kanade, T. (1997). A paraperspective factorization method for shape and motion recovery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(3), 206–219.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). *Numerical recipes in C* (2nd edn.). Cambridge: Cambridge University Press.

Ramamoorthi, R. (2002). Analytic PCA construction for theoretical analysis of lighting variability in images of a Lambertian object. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(10), 1322–1333.

Ramamoorthi, R., & Hanrahan, P. (2001). On the relationship between radiance and irradiance: Determining the illumination from images of a convex Lambertian object. *Journal of the Optical Society of America (JOSA A)*, *18*(10), 2448–2459.

Shashua, A., & Avidan, S. (1996). The rank 4 constraint in multiple (> 2) view geometry. In *Proc. European conf. computer vision* (pp. 196–206).

Tomasi, C., & Kanade, T. (1990). Shape and motion without depth. In *Proc. int'l conf. computer vision*.

Tomasi, C., & Kanade, T. (1992). Shape and motion from image streams under orthography: A factorization method. *International Journal of Computer Vision*, *9*(2), 137–154.

Triggs, B., McLauchlan, P., Hartley, R., & Fitzgibbon, A. (2000). Bundle adjustment—A modern synthesis. In W. Triggs, A. Zisserman, & R. Szeliski (Eds.), *LNCS. Vision algorithms: theory and practice* (pp. 298–375). Berlin: Springer.

Zelnik-Manor, L., & Irani, M. (1999a). Multi-frame alignment of planes. In *Proc. conf. computer vision and pattern recognition*.

Zelnik-Manor, L., & Irani, M. (1999b). Multi-view subspace constraints on homographies. In *Proc. int'l conf. computer vision* (pp. 710–715).

Zelnik-Manor, L., & Irani, M. (2002). Multi-view subspace constraints on homographies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(2), 214–223.