

A constrained parameter estimation approach for some vision problems

Wojciech Chojnacki, Michael J. Brooks, Anton van den Hengel, Darren Gawley

Department of Computer Science, University of Adelaide
Adelaide, SA 5005, Australia

{wojteck,mjb,hengel,dg}@cs.adelaide.edu.au

Abstract

Previous work of the authors developed a theoretically well-founded scheme (FNS) for finding the minimiser of a class of cost functions. Various problems in video analysis, stereo vision, ellipse-fitting, etc, may be expressed in terms of finding such a minimiser. However, in common with many other approaches, it is necessary to correct the minimiser as a post-process if an ancillary constraint is also to be satisfied. In this paper we develop the first integrated scheme (CFNS) for simultaneously minimising the cost function and satisfying the constraint. Preliminary experiments in the domain of fundamental-matrix estimation show that CFNS generates rank-2 estimates with smaller cost function values than rank-2 corrected FNS estimates. Furthermore, when compared with the Hartley-Zisserman Gold Standard method, CFNS is seen to generate results of comparable quality in a fraction of the time.

1. Introduction

Many problems in computer vision involve estimating the parameters that constrain a set of image feature locations. In some cases, the parameters are subject to *ancillary constraints* not involving feature locations. Two example problems of this form are the stereo and motion problems of estimating coefficients of the *epipolar equation* [4], and estimating coefficients of the *differential epipolar equation* [1], each involving an ancillary *cubic constraint*. The *principal equation* that applies to a wide class of problems, including those specified above, takes the form

$$\boldsymbol{\theta}^T \mathbf{u}(\mathbf{x}) = 0. \quad (1)$$

Here $\boldsymbol{\theta} = [\theta_1, \dots, \theta_l]^T$ is a vector representing unknown parameters; $\mathbf{x} = [x_1, \dots, x_k]^T$ is a vector representing an element of the data (for example, the lo-

cations of a pair of corresponding points); and $\mathbf{u}(\mathbf{x}) = [u_1(\mathbf{x}), \dots, u_l(\mathbf{x})]^T$ is a vector with the data transformed in a problem-dependent manner such that: (i) each component $u_i(\mathbf{x})$ is a quadratic form in the compound vector $[\mathbf{x}^T, 1]^T$, (ii) one component is equal to 1. A common form of the ancillary constraint is

$$\phi(\boldsymbol{\theta}) = 0, \quad (2)$$

where ϕ is a scalar-valued function *homogeneous* of degree κ —that is such that

$$\phi(t\boldsymbol{\theta}) = t^\kappa \phi(\boldsymbol{\theta}) \quad (3)$$

for every non-zero scalar t . The estimation problem can now be stated as follows: Given a collection $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ of *image data* and a meaningful *cost function* that characterises the extent to which any particular $\boldsymbol{\theta}$ fails to satisfy the system of the copies of equation (1) associated with $\mathbf{x} = \mathbf{x}_i$ ($i = 1, \dots, n$), find $\boldsymbol{\theta} \neq \mathbf{0}$ satisfying (2) for which the cost function attains its minimum. With the aid of the principle of maximum likelihood, a compelling case may be made for adoption of the cost function

$$J_{\text{AML}}(\boldsymbol{\theta}; \mathbf{x}_1, \dots, \mathbf{x}_n) = \sum_{i=1}^n \frac{\boldsymbol{\theta}^T \mathbf{u}(\mathbf{x}_i) \mathbf{u}(\mathbf{x}_i)^T \boldsymbol{\theta}}{\boldsymbol{\theta}^T \partial_{\mathbf{x}} \mathbf{u}(\mathbf{x}_i) \boldsymbol{\Lambda}_{\mathbf{x}_i} \partial_{\mathbf{x}} \mathbf{u}(\mathbf{x}_i)^T \boldsymbol{\theta}},$$

where, for any length k vector \mathbf{y} , $\partial_{\mathbf{x}} \mathbf{u}(\mathbf{y})$ denotes the $l \times k$ matrix of the partial derivatives of the function $\mathbf{x} \mapsto \mathbf{u}(\mathbf{x})$ evaluated at \mathbf{y} , and, for each $i = 1, \dots, n$, $\boldsymbol{\Lambda}_{\mathbf{x}_i}$ is a $k \times k$ symmetric *covariance matrix* describing the uncertainty of the data point \mathbf{x}_i (see [3, 8]). If J_{AML} is minimised over those non-zero parameter vectors for which (2) holds, then the vector at which the minimum of J_{AML} is attained, the *constrained minimum likelihood estimate* $\hat{\boldsymbol{\theta}}_{\text{AML}}$. The function $\boldsymbol{\theta} \mapsto J_{\text{AML}}(\boldsymbol{\theta}; \mathbf{x}_1, \dots, \mathbf{x}_n)$ is homogeneous of degree zero and the zero set of ϕ is unaffected by multiplication

by non-zero scalars, so $\hat{\boldsymbol{\theta}}_{\text{AML}}$ is determined only up to scale.

This paper presents a method for determining $\hat{\boldsymbol{\theta}}_{\text{AML}}$. Unlike previous approaches, the proposed method handles the ancillary constraint in an integrated fashion, not by means of a correction process operating upon results of unconstrained minimisation. The results of experiments in fundamental matrix estimation given here indicate that the method attains a very high level of performance in terms of accuracy and speed.

2. Fundamental numerical scheme

Isolating the constrained minimiser of J_{AML} is a challenging problem. Much easier to find is the *unconstrained approximated maximum likelihood estimate*, $\hat{\boldsymbol{\theta}}_{\text{AML}}^u$, defined as the *unconstrained* minimiser of J_{AML} ; it is obtained by ignoring the ancillary constraint and searching over all of the parameter space. While $\hat{\boldsymbol{\theta}}_{\text{AML}}^u$ cannot be expressed in closed form, a numerical approximation to it can be calculated by employing a suitable numerical scheme [3]. Underpinning numerical calculation is the fact that $\hat{\boldsymbol{\theta}}_{\text{AML}}^u$ satisfies the *variational equation* for unconstrained minimisation

$$[\partial_{\boldsymbol{\theta}} J_{\text{AML}}(\boldsymbol{\theta}; \mathbf{x}_1, \dots, \mathbf{x}_n)]_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_{\text{AML}}^u} = \mathbf{0}^T \quad (4)$$

with $\partial_{\boldsymbol{\theta}} J_{\text{AML}}$ the row vector of the partial derivatives of J_{AML} with respect to $\boldsymbol{\theta}$. Direct computation shows that

$$[\partial_{\boldsymbol{\theta}} J_{\text{AML}}(\boldsymbol{\theta}; \mathbf{x}_1, \dots, \mathbf{x}_n)]^T = 2\mathbf{X}_{\boldsymbol{\theta}}\boldsymbol{\theta}, \quad (5)$$

where

$$\mathbf{X}_{\boldsymbol{\theta}} = \sum_{i=1}^n \frac{\mathbf{A}_i}{\boldsymbol{\theta}^T \mathbf{B}_i \boldsymbol{\theta}} - \sum_{i=1}^n \frac{\boldsymbol{\theta}^T \mathbf{A}_i \boldsymbol{\theta}}{(\boldsymbol{\theta}^T \mathbf{B}_i \boldsymbol{\theta})^2} \mathbf{B}_i,$$

$$\mathbf{A}_i = \mathbf{u}(\mathbf{x}_i)\mathbf{u}(\mathbf{x}_i)^T, \quad \mathbf{B}_i = \partial_{\mathbf{x}}\mathbf{u}(\mathbf{x}_i)\boldsymbol{\Lambda}_{\mathbf{x}_i}\partial_{\mathbf{x}}\mathbf{u}(\mathbf{x}_i)^T.$$

Thus (4) can be written as

$$[\mathbf{X}_{\boldsymbol{\theta}}\boldsymbol{\theta}]_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_{\text{AML}}^u} = \mathbf{0}, \quad (6)$$

providing a convenient basis for determining $\hat{\boldsymbol{\theta}}_{\text{AML}}^u$. A straightforward algorithm for numerically solving this equation can be derived by realising that a vector $\boldsymbol{\theta}$ satisfies (6) if and only if it falls into the null space of the matrix $\mathbf{X}_{\boldsymbol{\theta}}$. Thus if $\boldsymbol{\theta}_{k-1}$ is a tentative approximate solution, then an improved solution can be obtained by picking a vector $\boldsymbol{\theta}_k$ from that eigenspace of $\mathbf{X}_{\boldsymbol{\theta}_{k-1}}$ which most closely approximates the null

space of $\mathbf{X}_{\boldsymbol{\theta}}$; this eigenspace is, of course, the one corresponding to the eigenvalue closest to zero in absolute value. The *fundamental numerical scheme* (FNS) implementing this idea is presented in Figure 1. The scheme is seeded with the *algebraic least squares* (ALS) *estimate*, $\hat{\boldsymbol{\theta}}_{\text{ALS}}$, defined as the unconstrained minimiser of the cost function $J_{\text{ALS}}(\boldsymbol{\theta}; \mathbf{x}_1, \dots, \mathbf{x}_n) = \|\boldsymbol{\theta}\|^{-2} \sum_{i=1}^n \boldsymbol{\theta}^T \mathbf{A}_i \boldsymbol{\theta}$. The estimate $\hat{\boldsymbol{\theta}}_{\text{ALS}}$ coincides, up to scale, with an eigenvector of $\sum_{i=1}^n \mathbf{A}_i$ associated with the smallest eigenvalue, and this can be found by performing singular-value decomposition (SVD) of the matrix $[\mathbf{u}(\mathbf{x}_1), \dots, \mathbf{u}(\mathbf{x}_n)]^T$.

1. Set $\boldsymbol{\theta}_0 = \hat{\boldsymbol{\theta}}_{\text{ALS}}$.
2. Assuming that $\boldsymbol{\theta}_{k-1}$ is known, compute the matrix $\mathbf{X}_{\boldsymbol{\theta}_{k-1}}$.
3. Compute a normalised eigenvector of $\mathbf{X}_{\boldsymbol{\theta}_{k-1}}$ corresponding to the eigenvalue closest to zero (in absolute value) and take this eigenvector for $\boldsymbol{\theta}_k$.
4. If $\boldsymbol{\theta}_k$ is sufficiently close to $\boldsymbol{\theta}_{k-1}$, then terminate the procedure; otherwise increment k and return to Step 2.

FIGURE 1. Fundamental numerical scheme.

Different but related schemes for numerically solving equations like (6) were developed by Leedan and Meer [9] and Matei and Meer [10]. Yet another approach is Kanatani's [8, Chap. 9] *renormalisation* scheme, in which an estimate is sought at which $\partial_{\boldsymbol{\theta}} J_{\text{AML}}$ is approximately zero (see [2] for details).

FNS has the advantages that it aims to minimise a cost function which is statistically well founded, provides a genuine means for theoretically calculating the minimiser, and is simply expressed and efficient. Furthermore, FNS proves useful in determining theoretical relationships between various existing methods of minimising J_{AML} (see [2]).

3. Constrained fundamental numerical scheme

With all its virtues, FNS lacks a proper accommodation of the ancillary constraint. A standard way of dealing with this problem is to adopt an adjustment procedure as a separate post-process (see [8, Chap. 5]). In contrast, as shown next, it is possible to merge FNS

and the ancillary constraint in a fully consistent, integrated fashion. The resulting scheme, the first of its kind, is a variant of FNS, in which \mathbf{X}_θ is replaced by a more complicated matrix.

The starting point of the development of the new algorithm is the variational equation for constrained minimisation

$$\begin{aligned} [\partial_\theta J_{\text{AML}}(\theta) + \lambda \partial_\theta \phi(\theta)]_{\theta=\hat{\theta}_{\text{AML}}} &= \mathbf{0}^T, \\ \phi(\hat{\theta}_{\text{AML}}) &= 0, \end{aligned} \quad (7)$$

where λ is a scalar *Lagrange multiplier*. Another crucial ingredient is the identity $\partial_\theta \phi(\theta)\theta = \kappa \phi(\theta)$ obtained by differentiating (3) with respect to t and evaluating at $t = 1$. With this identity, the equation $\phi(\theta) = 0$ is equivalent to $\mathbf{a}_\theta^T \theta = 0$, where $\mathbf{a}_\theta = [\partial_\theta \phi(\theta)]^T / 2$. Combining this with (5), the system (7) becomes

$$\mathbf{X}_\theta \theta + \lambda \mathbf{a}_\theta = \mathbf{0}, \quad (8)$$

$$\mathbf{a}_\theta^T \theta = 0, \quad (9)$$

where all evaluations at $\hat{\theta}_{\text{AML}}$ are dropped for clarity. Premultiplying both sides of (8) by \mathbf{a}_θ^T and bearing in mind that $\mathbf{a}_\theta^T \mathbf{a}_\theta = \|\mathbf{a}_\theta\|^2$, we find that

$$\mathbf{a}_\theta^T \mathbf{X}_\theta \theta + \|\mathbf{a}_\theta\|^2 \lambda = 0$$

whence $\lambda = -\|\mathbf{a}_\theta\|^{-2} \mathbf{a}_\theta^T \mathbf{X}_\theta \theta$. Consequently, (8) can be written as

$$(\mathbf{X}_\theta - \|\mathbf{a}_\theta\|^{-2} \mathbf{a}_\theta \mathbf{a}_\theta^T \mathbf{X}_\theta) \theta = \mathbf{0}. \quad (10)$$

Let \mathbf{P}_θ be the $l \times l$ matrix given by

$$\mathbf{P}_\theta = \mathbf{I}_l - \|\mathbf{a}_\theta\|^{-2} \mathbf{a}_\theta \mathbf{a}_\theta^T,$$

where \mathbf{I}_l denotes the $l \times l$ identity matrix. Note that \mathbf{P}_θ is symmetric and obeys $\mathbf{P}_\theta^2 = \mathbf{P}_\theta$. With use of \mathbf{P}_θ , (10) can be written as

$$\mathbf{P}_\theta \mathbf{X}_\theta \theta = \mathbf{0}. \quad (11)$$

In view of (9),

$$(\mathbf{I}_l - \mathbf{P}_\theta) \theta = \|\mathbf{a}_\theta\|^{-2} \mathbf{a}_\theta \mathbf{a}_\theta^T \theta = \mathbf{0}. \quad (12)$$

Hence $\mathbf{P}_\theta \theta = \theta$, and so (11) immediately leads to

$$\mathbf{P}_\theta \mathbf{X}_\theta \mathbf{P}_\theta \theta = \mathbf{0}. \quad (13)$$

Let \mathbf{Y}_θ be the $l \times l$ matrix defined by

$$\mathbf{Y}_\theta = \|\theta\|^2 \mathbf{P}_\theta \mathbf{X}_\theta \mathbf{P}_\theta + \mathbf{I}_l - \mathbf{P}_\theta.$$

Clearly, \mathbf{Y}_θ is symmetric. Since the function $\theta \mapsto \mathbf{P}_\theta$ is homogeneous of degree 0 and the function $\theta \mapsto \mathbf{X}_\theta$

is homogeneous of degree -2 , it follows that the function $\theta \mapsto \mathbf{Y}_\theta$ is consistently homogeneous of degree 0. In view of (12) and (13),

$$\mathbf{Y}_\theta \theta = \mathbf{0}. \quad (14)$$

It can be shown that this equation is actually equivalent to the system comprising (8) and (9).

At this stage, one would hope that a modified FNS with \mathbf{Y}_θ playing the role of \mathbf{X}_θ would be a good tool for calculating $\hat{\theta}_{\text{AML}}$. Unfortunately, this algorithm fails to converge.

As a remedy, we put forward a rather complex modification. The new algorithm retains the logic of the tentative one, but now \mathbf{Y}_θ is replaced by a more complicated matrix ensuring the convergence of the algorithm. The new matrix is derived from a matrix \mathbf{Z}_θ chosen so that (14) is equivalent to

$$\mathbf{Z}_\theta \theta = \mathbf{0}. \quad (15)$$

To define \mathbf{Z}_θ , we need a few preliminaries. Denote by \mathbf{H}_θ the Hessian of J_{AML} at θ . Direct if tedious calculation shows that

$$\mathbf{H}_\theta = 2(\mathbf{X}_\theta - \mathbf{T}_\theta),$$

where

$$\begin{aligned} \mathbf{T}_\theta = \sum_{i=1}^n \frac{2}{(\theta^T \mathbf{B}_i \theta)^2} & \left[\mathbf{A}_i \theta \theta^T \mathbf{B}_i + \mathbf{B}_i \theta \theta^T \mathbf{A}_i \right. \\ & \left. - 2 \frac{\theta^T \mathbf{A}_i \theta}{\theta^T \mathbf{B}_i \theta} \mathbf{B}_i \theta \theta^T \mathbf{B}_i \right]. \end{aligned}$$

Let Φ_θ be the Hessian of ϕ at θ . For each $i \in \{1, \dots, l\}$, let \mathbf{e}_i be the length l vector whose i th entry is unital and all other entries are zero. With these preparations, set

$$\mathbf{Z}_\theta = \mathbf{A}_\theta + \mathbf{B}_\theta + \mathbf{C}_\theta,$$

where

$$\mathbf{A}_\theta = \mathbf{P}_\theta \mathbf{H}_\theta (2\theta \theta^T - \|\theta\|^2 \mathbf{I}_l),$$

$$\mathbf{B}_\theta = \|\theta\|^2 \|\mathbf{a}_\theta\|^{-2}$$

$$\times \left[\sum_{i=1}^l (\Phi_\theta \mathbf{e}_i \mathbf{a}_\theta^T + \mathbf{a}_\theta \mathbf{e}_i^T \Phi_\theta) \mathbf{X}_\theta \theta \mathbf{e}_i^T \right.$$

$$\left. - 2 \|\mathbf{a}_\theta\|^{-2} \mathbf{a}_\theta \mathbf{a}_\theta^T \mathbf{X}_\theta \theta \mathbf{a}_\theta^T \Phi_\theta \right],$$

$$\mathbf{C}_\theta = \|\mathbf{a}_\theta\|^{-2} \kappa \left[\frac{\phi(\theta)}{4} \Phi_\theta + \mathbf{a}_\theta \mathbf{a}_\theta^T \right.$$

$$\left. - \frac{\phi(\theta)}{2} \|\mathbf{a}_\theta\|^{-2} \mathbf{a}_\theta \mathbf{a}_\theta^T \Phi_\theta \right].$$

Note that Z_θ is *not* symmetric. Direct calculation shows that the matrix functions $\theta \mapsto A_\theta$, $\theta \mapsto B_\theta$ and $\theta \mapsto C_\theta$ are such that $A_\theta\theta = \|\theta\|^2 P_\theta H_\theta \theta$, $B_\theta\theta = 0$ and $C_\theta\theta = (I_l - P_\theta)\theta$ for each θ . Therefore (15) is equivalent to

$$(\|\theta\|^2 P_\theta H_\theta + I_l - P_\theta)\theta = 0,$$

which is in turn equivalent to (14), as a simple argument shows.

Note that $Z_\theta\theta = 0$ is equivalent to

$$Q_\theta\theta = 0 \quad (16)$$

where $Q_\theta = Z_\theta^T Z_\theta$ is a symmetric matrix. This allows Q_θ to be taken for the ultimate replacement for X_θ . The steps of the resulting *constrained fundamental numerical scheme* (CFNS) are given in Figure 2.

1. Set $\theta_0 = \hat{\theta}_{\text{ALS}}$.
2. Assuming that θ_{k-1} is known, compute the matrix $Q_{\theta_{k-1}}$.
3. Compute a normalised eigenvector of $Q_{\theta_{k-1}}$ corresponding to the eigenvalue closest to zero (in absolute value) and take this eigenvector for θ_k .
4. If θ_k is sufficiently close to θ_{k-1} , then terminate the procedure; otherwise increment k and return to Step 2.

FIGURE 2. Constrained fundamental numerical scheme.

A necessary condition for CFNS to converge to a solution θ^* of (16) is that the smallest (non-negative) eigenvalue of Q_{θ^*} should be sufficiently well separated from the remaining eigenvalues. When this condition is satisfied, the algorithm seeded with an estimate close enough to θ^* will produce updates quickly converging to θ^* .

4. Experiments

We now apply our constrained estimation method to the problem of fundamental matrix computation. This may readily be expressed within our general problem form, the constraint being that the resulting matrix is singular.

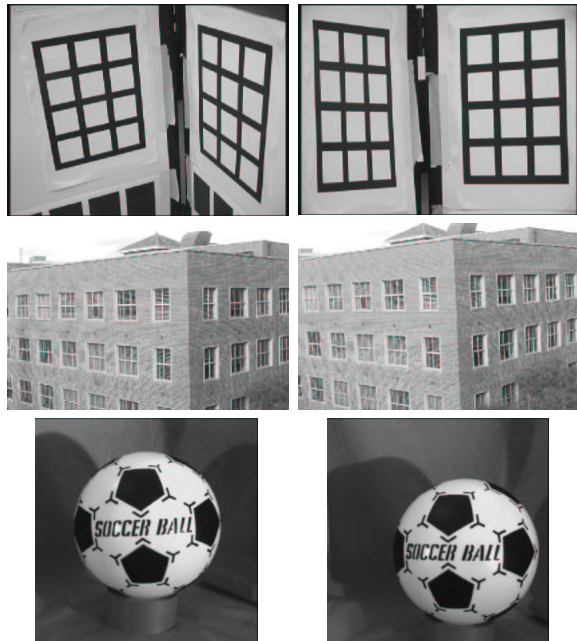


FIGURE 3. The grid, building, and soccer ball stereo image pairs.

Three different stereo image pairs were used, as shown in Figure 3. These pairs exhibit variation in subject matter, and in the camera set-up used for acquisition. Features were detected in each image using the Harris corner detector [5]. A set of corresponding points was generated for each pair by manually matching the detected features. The number of matched points was 96 for the grid, 44 for the building, and 55 for the soccer ball. For each estimation method, the entire set of matched points was used to compute a fundamental matrix. The covariances of the data were assumed to be default identity matrices corresponding to isotropic homogeneous noise in image point measurement.

The basic methods used were:

- **ALS** = Algebraic Least Squares
- **FNS** = Fundamental Numerical Scheme
- **CFNS** = Constrained FNS
- **GS** = Gold Standard Method

The FNS method has previously been shown to be operationally equivalent to using a minimisation procedure such as Levenberg–Marquardt’s directly on J_{AML} [3]. Prior to running ALS, FNS and CFNS, the data were normalised with the use of Hartley’s algorithm [6]. In our tests, this normalisation improved the

| | J_{AML} | $ \phi $ |
|-------|------------------|-------------------------|
| FNS | 4.876 | 1.617×10^{-11} |
| FNS+ | 9.493 | 0 |
| CFNS | 4.997 | 3.179×10^{-20} |
| CFNS+ | 4.997 | 0 |

TABLE 1. J_{AML} and $|\phi|$ values for FNS and CFNS before and after SVD rank-2 correction: grid images case.

performance of ALS markedly and had negligible impact on FNS. However, we observed that data normalisation is a critical requirement for the successful implementation of CFNS; the CFNS algorithm failed to converge when used with raw image data. (It emerges that normalisation of the data acts to improve separation of the smaller eigenvalues of the matrices \mathbf{Q}_θ involved.) GS is the (MLE) Hartley-Zisserman Gold Standard method [7] seeded with the FNS estimate.

The constraint that a fundamental matrix should have determinant zero (and hence have rank 2) is ignored by the ALS and FNS methods. Therefore their estimates are not generally usable without modification. Given a particular method M, we shall use the notation M+ to indicate that a direct rank-2 correction to the M estimate is made. An estimate $\hat{\mathbf{F}}$ with $\|\hat{\mathbf{F}}\|_F = 1$ is modified to a rank-2 matrix $\hat{\mathbf{F}}_c$ with $\|\hat{\mathbf{F}}_c\|_F = 1$ by minimising the distance $\|\hat{\mathbf{F}} - \hat{\mathbf{F}}_c\|_F$ subject to the condition $\det \hat{\mathbf{F}}_c = 0$, where $\|\cdot\|_F$ denotes the Frobenius norm. This is implemented simply by performing an SVD of the estimate, setting the smallest singular value to zero, and recomposing. We follow the advice of Hartley and perform this correction before back-transforming the estimate.

We additionally employ a more sophisticated rank-2 correction method to FNS that we denote FNS++. It is the composition of three methods: FNS; a more sophisticated, iterative, Kanatani-like correction method; and SVD correction (a final step to secure generation of perfectly rank-2 estimates). The iterative correction is obtained by means of the process

$$\begin{aligned} \boldsymbol{\theta}_{k+1} = & \boldsymbol{\theta}_k - [\partial_\theta \phi(\boldsymbol{\theta}_k) \mathbf{H}_{\boldsymbol{\theta}_k}^{-1} [\partial_\theta \phi(\boldsymbol{\theta}_k)]^T]^{-1} \\ & \times \phi(\boldsymbol{\theta}_k) \mathbf{H}_{\boldsymbol{\theta}_k}^{-1} [\partial_\theta \phi(\boldsymbol{\theta}_k)]^T, \end{aligned}$$

where $\mathbf{H}_{\boldsymbol{\theta}_k}^{-1}$ denotes the pseudo-inverse of $\mathbf{H}_{\boldsymbol{\theta}_k}$.

Each estimator was used to generate a fundamental matrix. The values of J_{AML} and the constraint residual $|\phi|$ are shown for various methods in Table 1. The intention here is to demonstrate the effects of

| | <i>Grid</i> | <i>Building</i> | <i>Soccer</i> |
|-------|-------------|-----------------|---------------|
| ALS+ | 15.06 | 5.347 | 0.799 |
| FNS+ | 9.493 | 4.883 | 0.813 |
| FNS++ | 4.997 | 2.052 | 0.442 |
| CFNS+ | 4.997 | 1.878 | 0.442 |
| GS+ | 5.014 | 1.879 | 0.442 |

TABLE 2. J_{AML} values for various rank-2 estimates.

| | <i>Grid</i> | <i>Building</i> | <i>Soccer</i> |
|-------|-------------|-----------------|---------------|
| ALS+ | 0.326 | 0.285 | 0.0926 |
| FNS+ | 0.263 | 0.275 | 0.0933 |
| FNS++ | 0.188 | 0.173 | 0.0681 |
| CFNS+ | 0.188 | 0.163 | 0.0681 |
| GS+ | 0.188 | 0.163 | 0.0681 |

TABLE 3. MLE cost function errors for various rank-2 estimates.

SVD rank-2 correction upon estimates obtained. It emerges that, although FNS provides estimates with low values of J_{AML} , the SVD rank-2 correction process (which renders usable fundamental matrices) considerably worsens the cost function values (ALS estimates also suffer this degradation after correction). In contrast CFNS generates estimates with low values of J_{AML} that are unaffected by subsequent rank-2 correction. In other words, CFNS+ considerably outperforms FNS+.

In Table 2 values of J_{AML} are given for the methods FNS+, FNS++, CFNS+, and GS+. Note that the ancillary constraint is in all cases perfectly satisfied. CFNS+ and GS+ give the best results and are essentially inseparable, while FNS++ is only slightly behind. FNS+ and ALS+ lag much further behind.

Similar results are presented in Table 3, except that comparison is made using the MLE (Gold Standard) residual. Once again, CFNS+ has a lower error than FNS+, a slightly lower error than FNS++, and is on par with the much more computationally intensive GS+.

We conclude by pointing out that results of synthetic image tests give a similar picture [11].

5. Conclusion

The J_{AML} cost function has been shown in many previous studies to be an important factor in obtain-

ing a first-order approximation to a maximum likelihood estimate. Earlier work of the authors showed that FNS is a theoretically well-founded and efficient method for finding the minimiser of J_{AML} , given an initial estimate in the neighbourhood of the global minimum. However, our experiments with fundamental matrix estimation show that SVD rank-2 correction of J_{AML} 's minimiser acts to considerably worsen the J_{AML} residual.

CFNS is a variant of the FNS method which incorporates constraint in an integrated manner, aiming simultaneously to minimise J_{AML} and a constraint-based expression. Our experiments suggest that CFNS achieves a smaller value of J_{AML} than is attained by either SVD rank-2 corrected FNS or SVD rank-2 corrected ALS. It produces only slightly smaller error than the slower, iteratively rank-2 corrected FNS, but has the advantage of being a conceptually sounder, integrated method for constrained minimisation. Furthermore, when compared with the much slower FNS-seeded GS, it gives almost identical results, both in terms of J_{AML} residual and GS's MLE cost function residual.

References

- [1] M. J. Brooks, W. Chojnacki, and L. Baumela. Determining the egomotion of an uncalibrated camera from instantaneous optical flow. *J. Optical Soc. Am. A*, 14(10):2670–2677, 1997.
- [2] W. Chojnacki, M. J. Brooks, and A. van den Hengel. Rationalising the renormalisation method of Kanatani. *J. Math. Imaging and Vision*, 14(1):21–38, 2001.
- [3] W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley. On the fitting of surfaces to data with covariances. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(11):1294–1303, 2000.
- [4] O. D. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. The MIT Press, Cambridge, Mass., 1993.
- [5] C. G. Harris and M. Stephens. A combined corner and edge detector. In *Proc. 4th Alvey Vision Conf.*, pages 147–151, 1988.
- [6] R. Hartley. In defense of the eight-point algorithm. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(6):580–593, 1997.
- [7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [8] K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice*. Elsevier, Amsterdam, 1996.
- [9] Y. Leedan and P. Meer. Heteroscedastic regression in computer vision: problems with bilinear constraint. *Int. J. Computer Vision*, 37(2):127–150, 2000.
- [10] B. Matei and P. Meer. A general method for errors-in-variables problems in computer vision. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, volume 2, pages 18–25, 2000.
- [11] A. van den Hengel, W. Chojnacki, M. J. Brooks, and D. Gawley. A new constrained parameter estimator: experiments in fundamental matrix computation. In *Proc. 13th British Machine Vision Conf.*, volume 2, pages 468–476, 2002.