A Decoupled Algorithm for Vision Parameter Estimation with Application to the Trifocal Tensor

Tony Scoleri[a,b], Wojciech Chojnacki[b], Michael J. Brooks[b]

[a]Defence Science and Technology Organisation, Edinburgh, SA 5111, Australia

[b]School of Computer Science, The University of Adelaide, Adelaide, SA 5005, Australia

tony.scoleri@dsto.defence.gov.au, {wojciech.chojnacki, michael.brooks}@adelaide.edu.au

Abstract

We consider the problem of estimating parameters of a model described by a system of equations which underlies a wide class of computer vision applications. One method to solve such a problem is the fundamental numerical scheme (FNS) previously proposed by some of the authors. In this paper, a more stable version of FNS is developed, with better convergence properties than the original version. The improvement in performance is achieved by reducing the original estimation problem to a couple of problems of lower dimension. By way of example, the new algorithm has been applied to the problem of estimating the trifocal tensor relating three views of a scene. Experiments carried out with both synthetic and real images reveal the new estimator to be more stable compared to the original FNS method, and commensurate in accuracy with the Gold Standard maximum likelihood estimator.

1. Introduction

Fitting parametric models to data is a ubiquitous task in computer vision. In a typical model, parameters describe a relationship among image feature locations. The parameters and image data pertaining to the model are combined in a system of equations

$$\mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) = \mathbf{0},$$

where $\mathbf{x}$ is a length-k vector describing an *ideal* data point, $\boldsymbol{\theta}$ is a length-l vector of parameters, and $\mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) = [f_1(\mathbf{x}, \boldsymbol{\theta}), \ldots, f_m(\mathbf{x}, \boldsymbol{\theta})]^\mathsf{T}$ is a vector of *multi-objective* constraints satisfying $f_i(\mathbf{x}, \boldsymbol{\theta}) = \mathbf{u}_i(\mathbf{x})^\mathsf{T}\boldsymbol{\theta}$, where $\mathbf{u}_i(\mathbf{x})$ is a length-l vector of polynomial functions in $[\mathbf{x}^\mathsf{T}, 1]^\mathsf{T}$; in this case the constraint vector can be succinctly written as

$$\mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) = \mathtt{U}(\mathbf{x})^\mathsf{T}\boldsymbol{\theta}, \tag{1}$$

where $\mathtt{U}(\mathbf{x}) = [\mathbf{u}_1(\mathbf{x}), \ldots, \mathbf{u}_m(\mathbf{x})]$ is a $l \times m$ *data carrier matrix*. Examples of models of this type involving a constraint vector with several components include a model encoding an image-based *homography* [16], and analogous models defining *trifocal* and *quadrifocal tensors* [4, 5, 14]. If *observed* data points $\mathbf{x}_1, \ldots, \mathbf{x}_n$ come equipped with *covariance* matrices $\Lambda_{\mathbf{x}_1}, \ldots, \Lambda_{\mathbf{x}_n}$ quantifying measurement errors in the data, then a statistically meaningful estimate of $\boldsymbol{\theta}$ based on the compound set of the data points and their covariances can be obtained by minimising the multi-objective *approximated maximum likelihood* (AML) cost function

$$J_{\mathrm{AML}}(\boldsymbol{\theta}; \mathbf{x}_1, \ldots, \mathbf{x}_n) = \sum_{i=1}^{n} \mathbf{f}(\mathbf{x}_i, \boldsymbol{\theta})^\mathsf{T} \Sigma(\mathbf{x}_i, \boldsymbol{\theta})^{-1} \mathbf{f}(\mathbf{x}_i, \boldsymbol{\theta}),$$

where $\Sigma(\mathbf{x}_i, \boldsymbol{\theta}) = \partial_{\mathbf{x}}\mathbf{f}(\mathbf{x}_i, \boldsymbol{\theta}) \Lambda_{\mathbf{x}_i} \partial_{\mathbf{x}}\mathbf{f}(\mathbf{x}_i, \boldsymbol{\theta})^\mathsf{T}$. Importantly, when the length m of the $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\theta})$ surpasses the common codimension r of the submanifolds of the form $\{\mathbf{x} \in \mathbb{R}^k \mid \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) = \mathbf{0}\}$ with $\boldsymbol{\theta}$ representing *ideal* parameters that might have generated the data, the inverses $\Sigma(\mathbf{x}_i, \boldsymbol{\theta})^{-1}$ in the above expression must be replaced by the *r-truncated pseudoinverses* $\Sigma(\mathbf{x}_i, \boldsymbol{\theta})_r^+$ [3, 7]. The approximated maximum likelihood *estimate* of $\boldsymbol{\theta}$, defined as the minimiser of $J_{\mathrm{AML}}(\bullet; \mathbf{x}_1, \ldots, \mathbf{x}_n)$ and denoted $\widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}}$, exhibits a nearly optimal statistical behaviour and—unlike the maximum likelihood estimate—is relatively inexpensive to compute [13]. Once $\widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}}$ has been generated, additional constraints—if they apply—involving the parameters alone can be accommodated via an adjustment procedure. In what follows we shall confine our attention to the estimation phase that precedes adjustment, concentrating effectively on unconstrained minimisation of J_{AML} (this is underlined by attaching the superscript "u" in the symbol for the AML estimate).

Various methods are available for finding $\widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}}$. One is the *fundamental numerical scheme* (FNS) introduced in [1] and extended in [13]. Another is the *heteroscedastic errors-in-variables* (HEIV) *scheme* proposed by Leedan and Meer [8] and further developed by Matei and Meer [12]. Both techniques estimate the parameters of the model iteratively. To ensure convergence, the methods often require a good

0-7695-3067-2/07 $25.00 © 2007 Crown Copyright
DOI 10.1109/DICTA.2007.21

initial parameter estimate, but sometimes even an accurate seed leads to divergence if the level of noise in the data is too high.

The main purpose of this paper is to present a reduced form of FNS, where only a subset of the total parameter vector is estimated iteratively and the remaining parameters are recovered in a single step based on the result of the earlier iterative process. The reduced algorithm in effect replaces the original estimation problem with a couple of problems of lower dimension. The algorithm is an extension to the multi-objective setting of the reduced FNS in the single-objective case given in [2]. As is demonstrated in the experimental section of the paper, the process of dimension reduction leads to significant benefits. Compared to the full form [13], the reduced form of the algorithm requires a less accurate initial estimate and enjoys better convergence properties. While the paper is primarily concerned with FNS, the optimality condition which underlies the reduced form of this algorithm can readily be exploited to advance a reduced form of HEIV.

2. Variational equation

The minimiser $\widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}}$ satisfies the necessary optimality condition

$$[\partial_{\boldsymbol{\theta}} J_{\mathrm{AML}}(\boldsymbol{\theta}; \mathbf{x}_1, \ldots, \mathbf{x}_n)]_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}}} = \mathbf{0}^\mathsf{T} \tag{2}$$

with $\partial_{\boldsymbol{\theta}} J_{\mathrm{AML}}$ the row vector of the partial derivatives of J_{AML} with respect to $\boldsymbol{\theta}$. We term this the *variational equation*. With the aid of (1) reformulated as

$$\mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) = (\boldsymbol{\theta}^\mathsf{T} \otimes \mathtt{I}_m)\,\mathrm{vec}(\mathtt{U}^\mathsf{T}), \tag{3}$$

where $\mathtt{I}_m$ denotes the $m \times m$ identity matrix, and vec and $\otimes$ denote the vectorisation and Kronecker product operators [9], respectively, it can be shown that

$$[\partial_{\boldsymbol{\theta}} J_{\mathrm{AML}}(\boldsymbol{\theta}; \mathbf{x}_1, \ldots, \mathbf{x}_n)]^\mathsf{T} = 2\mathtt{X}_{\boldsymbol{\theta}}\boldsymbol{\theta},$$

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

$$\mathtt{M}_{\boldsymbol{\theta}} = \sum_{i=1}^{n} \mathtt{U}_i \Sigma_i^{-1} \mathtt{U}_i^\mathsf{T}, \tag{4a}$$

$$\mathtt{N}_{\boldsymbol{\theta}} = \sum_{i=1}^{n} (\mathtt{I}_l \otimes \boldsymbol{\eta}_i^\mathsf{T}) \mathtt{B}_i (\mathtt{I}_l \otimes \boldsymbol{\eta}_i), \tag{4b}$$

$$\begin{aligned}
\mathtt{U}_i &= \mathtt{U}(\mathbf{x}_i) = [\mathbf{u}_1(\mathbf{x}_i), \ldots, \mathbf{u}_m(\mathbf{x}_i)], \\
\mathtt{B}_i &= \partial_{\mathbf{x}_i} \mathrm{vec}(\mathtt{U}_i^\mathsf{T}) \Lambda_{\mathbf{x}_i} [\partial_{\mathbf{x}_i} \mathrm{vec}(\mathtt{U}_i^\mathsf{T})]^\mathsf{T}, \\
\Sigma_i &= (\boldsymbol{\theta}^\mathsf{T} \otimes \mathtt{I}_m) \mathtt{B}_i (\boldsymbol{\theta} \otimes \mathtt{I}_m), \qquad (4c) \\
\boldsymbol{\eta}_i &= \Sigma_i^{-1} \mathtt{U}_i^\mathsf{T} \boldsymbol{\theta}.
\end{aligned}$$

The variational equation (2) can accordingly be rewritten as

$$\mathtt{X}_{\boldsymbol{\theta}} \boldsymbol{\theta} = \mathbf{0}, \tag{5}$$

where the evaluation at $\widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}}$ is dropped for clarity. In this form the variational equation will serve as a basis for isolating $\widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}}$. As a consequence of $J_{\mathrm{AML}}(\bullet; \mathbf{x}_1, \ldots, \mathbf{x}_n)$ being homogeneous of degree zero, $\widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}}$ will only be determined up to scale.

If in calculating $(\partial_{\boldsymbol{\theta}} J_{\mathrm{AML}})^\mathsf{T}$ the identity

$$\mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) = (\mathtt{I}_m \otimes \boldsymbol{\theta}^\mathsf{T})\,\mathrm{vec}(\mathtt{U}) \tag{6}$$

is used instead of (3), then the ensuing expression for $\mathtt{M}_{\boldsymbol{\theta}}$ will be identical with the one given in (4a), but the expression for $\mathtt{N}_{\boldsymbol{\theta}}$ will change to take the form

$$\begin{aligned}
\mathtt{N}_{\boldsymbol{\theta}} &= \sum_{i=1}^{n} (\boldsymbol{\eta}_i^\mathsf{T} \otimes \mathtt{I}_l) \mathtt{B}_i^* (\boldsymbol{\eta}_i \otimes \mathtt{I}_l), \\
\mathtt{B}_i^* &= \partial_{\mathbf{x}_i} \mathrm{vec}(\mathtt{U}_i) \Lambda_{\mathbf{x}_i} [\partial_{\mathbf{x}_i} \mathrm{vec}(\mathtt{U}_i)]^\mathsf{T}, \\
\Sigma_i^* &= (\mathtt{I}_m \otimes \boldsymbol{\theta}^\mathsf{T}) \mathtt{B}_i^* (\mathtt{I}_m \otimes \boldsymbol{\theta}), \\
\boldsymbol{\eta}_i &= \Sigma_i^{*-1} \mathtt{U}_i^\mathsf{T} \boldsymbol{\theta}.
\end{aligned}$$

While the latter equation for $\mathtt{N}_{\boldsymbol{\theta}}$ arises more frequently in the literature [12], equation (4b) will prove more useful in what follows.

3. Reduced variational equation

Assume that the carrier matrix $\mathtt{U}(\mathbf{x})$ can be written as

$$\mathtt{U}(\mathbf{x}) = \begin{bmatrix} \mathtt{Z}(\mathbf{x}) \\ \mathtt{W} \end{bmatrix} = \begin{bmatrix} \mathbf{z}_1(\mathbf{x}) & \ldots & \mathbf{z}_m(\mathbf{x}) \\ \mathbf{w}_1 & \ldots & \mathbf{w}_m \end{bmatrix}, \tag{7}$$

where $\mathtt{Z}(\mathbf{x})$ is a $(l-m) \times m$ matrix that depends on $\mathbf{x}$ (a "pure measurement" matrix), and $\mathtt{W}$ is a $m \times m$ *invertible* matrix that does not depend on $\mathbf{x}$. Corresponding to this splitting of $\mathtt{U}(\mathbf{x})$, the parameter vector $\boldsymbol{\theta}$ will be subdivided as

$$\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\alpha} \end{bmatrix}, \tag{8}$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\alpha}$ are vectors of length $l-m$ and m respectively. The partitioning of $\mathtt{U}(\mathbf{x})$ and $\boldsymbol{\theta}$ reflects that fact that some components of $\boldsymbol{\theta}$, considered as indeterminates, appear in each of the equations of (1) only with constant coefficients. The vector $\boldsymbol{\alpha}$ collects together those components of $\boldsymbol{\theta}$ that appear in (1) with pure constant coefficients. For each $i = 1, \ldots, m$, the non-zero entries of the vector $\mathbf{w}_i$ represent the constant coefficients of the components of $\boldsymbol{\alpha}$ in the ith equation of (1). If, for instance, every equation of (1) has exactly one parameter with a unity coefficient, then, upon possible reordering of the equations of (1), we obtain $\mathtt{W} = \mathtt{I}_m$.

We shall now present a system of two equations that jointly are equivalent to the variational equation (5). One

of these equations involves only $\boldsymbol{\mu}$ and can be solved separately, and the other expresses $\boldsymbol{\alpha}$ in terms of $\boldsymbol{\mu}$. We begin by noting that, in view of (7),

$$\mathrm{vec}(\mathtt{U}(\mathbf{x})^{\mathsf{T}}) = \begin{bmatrix} \mathrm{vec}(\mathtt{Z}(\mathbf{x})^{\mathsf{T}}) \\ \mathrm{vec}(\mathtt{W}^{\mathsf{T}}) \end{bmatrix}$$

so that

$$\partial_{\mathbf{x}}\mathrm{vec}(\mathtt{U}(\mathbf{x})^{\mathsf{T}}) = \begin{bmatrix} \partial_{\mathbf{x}}\mathrm{vec}(\mathtt{Z}(\mathbf{x})^{\mathsf{T}}) \\ \mathtt{O}_{m^2\times k} \end{bmatrix}.$$

Hence, for each $i = 1, \ldots, n$, the $lm \times lm$ matrix $\mathtt{B}_i$ can be represented as

$$\mathtt{B}_i = \begin{bmatrix} \mathtt{B}_i^0 & \mathtt{O}_{(l-m)m\times m^2} \\ \mathtt{O}_{m^2\times(l-m)m} & \mathtt{O}_{m^2\times m^2} \end{bmatrix}, \tag{9}$$

where $\mathtt{B}_i^0$ is the $(l-m)m \times (l-m)m$ matrix given by

$$\mathtt{B}_i^0 = \partial_{\mathbf{x}_i}\mathrm{vec}(\mathtt{Z}_i^{\mathsf{T}})\Lambda_{\mathbf{x}_i}[\partial_{\mathbf{x}_i}\mathrm{vec}(\mathtt{Z}_i^{\mathsf{T}})]^{\mathsf{T}}, \quad \mathtt{Z}_i = \mathtt{Z}(\mathbf{x}_i).$$

It is worth noting that this partitioning of $\mathtt{B}_i$, crucial to the subsequent development, results from taking (3) rather than (6) as a point of departure.

Define a $m \times m$ matrix Σ_i' by

$$\Sigma_i' = (\boldsymbol{\mu}^{\mathsf{T}} \otimes \mathtt{I}_m)\mathtt{B}_i^0(\boldsymbol{\mu} \otimes \mathtt{I}_m). \tag{10}$$

Clearly, Σ_i' is positive semidefinite and depends only on the ith element of data, its covariance $\Lambda_{\mathbf{x}_i}$, and the parameter vector $\boldsymbol{\mu}$. Assume henceforth that each Σ_i' is positive definite and hence invertible. The inverses $\Sigma_i'^{-1}$ can now be used as matricial weights to define a "centroid" of the $\mathtt{Z}_i$ as follows:

$$\tilde{\mathtt{Z}} = \sum_{i=1}^{n} \mathtt{Z}_i\Sigma_i'^{-1}\Big(\sum_{i=1}^{n}\Sigma_i'^{-1}\Big)^{-1}.$$

Here $\sum_{i=1}^n \Sigma_i'^{-1}$ is invertible because a sum of positive definite matrices is also positive definite. For each $i = 1, \ldots, n$, let $\mathtt{Z}_i' = \mathtt{Z}_i - \tilde{\mathtt{Z}}$ be the ith pure measurement vector relative to $\tilde{\mathtt{Z}}$. Letting $\boldsymbol{\eta}_i' = \Sigma_i'^{-1}\mathtt{Z}_i'^{\mathsf{T}}\boldsymbol{\mu}$, define the following $(l-m)\times(l-m)$ matrices

$$\begin{aligned} \mathtt{M}_{\boldsymbol{\mu}}' &= \sum_{i=1}^{n} \mathtt{Z}_i'\Sigma_i'^{-1}\mathtt{Z}_i'^{\mathsf{T}}, \\ \mathtt{N}_{\boldsymbol{\mu}}' &= \sum_{i=1}^{n} (\mathtt{I}_{l-m} \otimes \boldsymbol{\eta}_i'^{\mathsf{T}})\mathtt{B}_i^0(\mathtt{I}_{l-m} \otimes \boldsymbol{\eta}_i'), \\ \mathtt{X}_{\boldsymbol{\mu}}' &= \mathtt{M}_{\boldsymbol{\mu}}' - \mathtt{N}_{\boldsymbol{\mu}}'. \end{aligned}$$

A fundamental result that can now be established is that $\boldsymbol{\theta} = [\boldsymbol{\mu}^{\mathsf{T}}, \boldsymbol{\alpha}^{\mathsf{T}}]^{\mathsf{T}}$ satisfies the variational equation (5) if and only if the following system of equations holds:

$$\mathtt{X}_{\boldsymbol{\mu}}'\boldsymbol{\mu} = \mathbf{0}, \tag{11a}$$

$$\boldsymbol{\alpha} = -(\tilde{\mathtt{Z}}\mathtt{W}^{-1})^{\mathsf{T}}\boldsymbol{\mu}. \tag{11b}$$

The first equation constrains solely $\boldsymbol{\mu}$ and, therefore, can be solved separately. Once $\boldsymbol{\mu}$ is determined, $\boldsymbol{\alpha}$ is readily prescribed by the second equation. Of the two constraints, the first plays a leading role and will be termed the *reduced variational equation*.

If one defines the *reduced AML cost function* by

$$J_{\mathrm{AML}}'(\boldsymbol{\mu}; \mathbf{x}_1, \ldots, \mathbf{x}_n) = \sum_{i=1}^{n} \boldsymbol{\mu}^{\mathsf{T}}\mathtt{Z}_i'\Sigma_i'^{-1}\mathtt{Z}_i'^{\mathsf{T}}\boldsymbol{\mu},$$

then (11a) gains the meaning of the variational equation for an optimiser of J_{AML}'. Interestingly, the $\boldsymbol{\mu}$-part of $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u$, which satisfies (11a) as $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u$ satisfies (5), turns out to be the minimiser of J_{AML}', denoted $\widehat{\boldsymbol{\mu}}_{\mathrm{AML}}^u$, not just a critical point of J_{AML}'. Moreover, both J_{AML} and J_{AML}' attain a common minimum value at $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u$ and $\widehat{\boldsymbol{\mu}}_{\mathrm{AML}}^u$, respectively.

4. Fundamental numerical scheme: full and reduced forms

A vector $\boldsymbol{\theta}$ satisfies (5) if and only if it is a solution of the *ordinary* eigenvalue problem $\mathtt{X}_{\boldsymbol{\theta}}\boldsymbol{\xi} = \lambda\boldsymbol{\xi}$, corresponding to the eigenvalue $\lambda = 0$. This suggests an iterative method for solving (5) whereby if $\boldsymbol{\theta}_{k-1}$ is a current approximate solution, then an updated solution $\boldsymbol{\theta}_k$ is a vector chosen from that eigenspace of $\mathtt{X}_{\boldsymbol{\theta}_{k-1}}$ which most closely approximates the null space of $\mathtt{X}_{\boldsymbol{\theta}}$; this eigenspace is, of course, the one corresponding to the eigenvalue closest to zero in absolute value. The process can be seeded with the *normalised algebraic least squares* (NALS) *estimate*, $\widehat{\boldsymbol{\theta}}_{\mathrm{NALS}}$. This estimate results from operating the *algebraic least-squares* (ALS) method on Hartley-normalised data [6]. ALS is a simple technique that computes the unconstrained minimiser of the cost function $J_{\mathrm{ALS}}(\boldsymbol{\theta}) = \|\boldsymbol{\theta}\|^{-2}\sum_{i=1}^n \boldsymbol{\theta}^T\mathtt{U}_i\mathtt{U}_i^T\boldsymbol{\theta}$, with $\|\boldsymbol{\theta}\| = (\sum_{j=1}^l \theta_j^2)^{1/2}$, by performing singular value decomposition on $[\mathtt{U}_1, \ldots, \mathtt{U}_n]^T$. The overall procedure constitutes the *fundamental numerical scheme* (FNS) [1] and is summarised in Figure 1.

1. Set $\boldsymbol{\theta}_0 = \widehat{\boldsymbol{\theta}}_{\mathrm{NALS}}$.
2. Assuming $\boldsymbol{\theta}_{k-1}$ is known, compute the matrix $\mathtt{X}_{\boldsymbol{\theta}_{k-1}}$.
3. Compute a normalised eigenvector of $\mathtt{X}_{\boldsymbol{\theta}_{k-1}}$ 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. Steps of the fundamental numerical scheme.

A modification of this technique based on the reduced variational system (11a) and (11b) is the *reduced fundamen-*

tal numerical scheme (RFNS). Its steps are given in Figure 2.

1. Set $\boldsymbol{\mu}_0 = \widehat{\boldsymbol{\mu}}_{\mathrm{NALS}}$.
2. Assuming $\boldsymbol{\mu}_{k-1}$ is known, compute the matrix $\mathtt{X}'_{\boldsymbol{\mu}_{k-1}}$.
3. Compute a normalised eigenvector of $\mathtt{X}'_{\boldsymbol{\mu}_{k-1}}$ corresponding to the eigenvalue closest to zero (in absolute value) and take this eigenvector for $\boldsymbol{\mu}_k$.
4. If $\boldsymbol{\mu}_k$ is sufficiently close to $\boldsymbol{\mu}_{k-1}$, then terminate the procedure; otherwise increment k and return to Step 2.
5. Compute $\boldsymbol{\alpha}$ as per (11b) using the limiting value $\boldsymbol{\mu}_k$ and the corresponding value $\tilde{\mathtt{Z}}(\boldsymbol{\mu}_k)$ from the previous step.

Figure 2. Steps of the reduced fundamental numerical scheme.

In the case that the matrices Σ_i^- are replaced by the matrices $(\Sigma_i)_r^+$ in the expression for J_{AML}, a similar change also affects the matrices $\mathtt{X}_{\boldsymbol{\theta}_k}$ of FNS. Moreover, recalling definitions (4c) and (10), and in view of (8) and (9), it follows that $\Sigma_i = \Sigma'_i$ for $i = 1, \ldots, n$, so the $(\Sigma'_i)_r^+$ supercede the $\Sigma_i'^{-1}$ in the expression for J'_{AML} and in the $\mathtt{X}'_{\boldsymbol{\mu}_k}$ of RFNS.

Finally, we remark that a vector $\boldsymbol{\theta}$ satisfying (5) can alternatively be viewed as a solution of the *generalised* eigenvalue problem $\mathtt{M}_{\boldsymbol{\theta}}\boldsymbol{\xi} = \lambda \mathtt{N}_{\boldsymbol{\theta}}\boldsymbol{\xi}$ corresponding to the eigenvalue $\lambda = 1$. This observation provides a starting point for the development of the *heteroscedastic-errors-in-variable* (HEIV) scheme in both full and reduced versions [2]. Each version solves successively generalised eigenvalue problems analogous to the ordinary eigenvalue problems solved by a corresponding version of FNS.

5. Experimental evaluation

We now present the results of comparative tests carried out to evaluate the performance of RFNS. The application considered is trifocal tensor estimation from point correspondences. The following five algorithms were used to compute tensors from both synthetic and real image data:

- NALS = Normalised Algebraic Least-Squares,
- HEIV = Heteroscedastic Errors-In-Variables scheme,
- FNS = Fundamental Numerical Scheme,
- RFNS = Reduced FNS,
- GS = Gold Standard.

GS is an advanced method [5] for minimising the *maximum likelihood* cost function J_{ML} given in (12) below. For fair comparison, FNS, RFNS, and GS are seeded with the *generalised total least-squares* (GTLS) tensor estimate [8, 11] in order to match HEIV given in [11] and all methods have similar termination conditions. A post-correction was applied to the final tensors obtained by FNS and RFNS to enforce internal constraints, see [11]. HEIV estimates were obtained by using the binary application supplied by the authors of the original HEIV papers [10].

5.1. Trifocal tensor from point matches

A trifocal tensor of three views is an analogue of a fundamental matrix of two views. It encapsulates all the geometric relations between three views that are independent of scene structure, but is more useful than the fundamental matrix as it enters constraints not only on point correspondences but also on line and combined point-and-line correspondences across the images.

Let $\mathtt{P} = [\mathtt{I}_3, \mathbf{0}]$, $\mathtt{P}' = [\mathtt{A}, \mathbf{e}'] = [a_i^j]$, and $\mathtt{P}'' = [\mathtt{B}, \mathbf{e}''] = [b_i^k]$ be three camera matrices, with $\mathtt{A}$ and $\mathtt{B}$ 3×3 matrices describing the infinite homographies from the first to the second and third images, respectively, and $\mathbf{e}'$ and $\mathbf{e}''$ the epipoles in the latter two views. The trifocal tensor is the valence-3 tensor, with one covariant and two contravariant indices, given by

$$\mathcal{T}_i^{jk} = a_i^j b_4^k - a_4^j b_i^k, \qquad i, j, k = 1, 2, 3.$$

Let $\mathbf{m} = [m^1, m^2, m^3]^\mathsf{T}$, $\mathbf{m}' = [m'^1, m'^2, 1]^\mathsf{T}$, and $\mathbf{m}'' = [m''^1, m''^2, 1]^\mathsf{T}$ be the images of a point $\mathbf{M}$ in 3D space, taken from the cameras with corresponding superscripts. The points are related through the trifocal tensor by the four trilinear constraints [14]:

$$\sum_{i=1}^{3}(m^i\mathcal{T}_i^{11} - m^i m'^1\mathcal{T}_i^{31} + m^i m'^1 m''^1\mathcal{T}_i^{33} - m^i m''^1\mathcal{T}_i^{13}) = 0,$$

$$\sum_{i=1}^{3}(m^i\mathcal{T}_i^{12} - m^i m'^1\mathcal{T}_i^{32} + m^i m'^1 m''^2\mathcal{T}_i^{33} - m^i m''^2\mathcal{T}_i^{13}) = 0,$$

$$\sum_{i=1}^{3}(m^i\mathcal{T}_i^{21} - m^i m'^2\mathcal{T}_i^{31} + m^i m'^2 m''^1\mathcal{T}_i^{33} - m^i m''^1\mathcal{T}_i^{23}) = 0,$$

$$\sum_{i=1}^{3}(m^i\mathcal{T}_i^{22} - m^i m'^2\mathcal{T}_i^{32} + m^i m'^2 m''^2\mathcal{T}_i^{33} - m^i m''^2\mathcal{T}_i^{23}) = 0.$$

Letting $m^3 = 1$, this system can be brought into the form given in (1) by first concatenating the inhomogeneous coordinates of $\mathbf{m}$, $\mathbf{m}'$, and $\mathbf{m}''$ to obtain a single item of data $\mathbf{x} = [m^1, m^2, m'^1, m'^2, m''^1, m''^2]$, next rearranging the tensor entries into a length-27 vector $\boldsymbol{\theta}$, and then setting $\mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) = [f_1(\mathbf{x}, \boldsymbol{\theta}), \ldots, f_4(\mathbf{x}, \boldsymbol{\theta})]^\mathsf{T}$, where $f_1, \ldots, f_4$ are the corresponding expressions on the left-hand side of the above system. Furthermore, $\mathtt{U}(\mathbf{x})$ and $\boldsymbol{\theta}$ can be partitioned as in (7) and (8) with $\mathtt{W} = \mathtt{I}_4$ and $\boldsymbol{\alpha} = [\mathcal{T}_3^{11}, \mathcal{T}_3^{12}, \mathcal{T}_3^{21}, \mathcal{T}_3^{22}]^\mathsf{T}$, respectively. Given a data set $\{\mathbf{x}_i\}_{i=1}^n$, the covariance of each datum $\mathbf{x}_i$ is assumed to be the default 6×6 identity matrix corresponding to isotropic homogeneous noise in image point measurement. Lastly, since the submanifolds of the form $\{\mathbf{x} \in \mathbb{R}^6 \mid \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) = \mathbf{0}\}$, with $\boldsymbol{\theta}$ representing a genuine

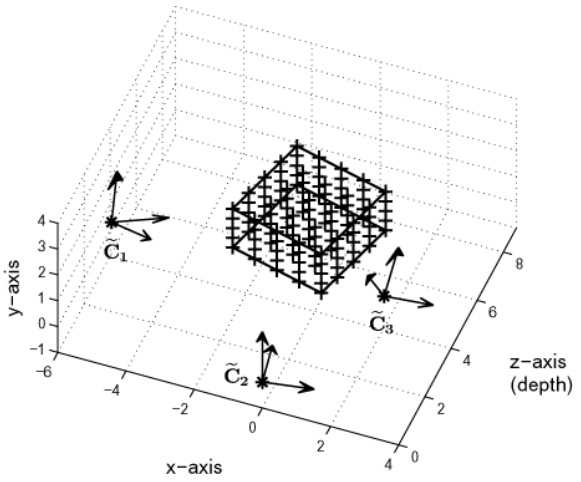


Figure 3. A synthetic 3D scene made of equally spaced points inside a cuboid and three cameras viewing the scene.

trifocal tensor, have codimension 3 [15], the matrices $(\Sigma_i)_3^+$ and $(\Sigma_i')_3^+$ must be used instead of the matrices Σ_i^{-1} and $\Sigma_i'^{-1}$ in the expressions for J_{AML} and J'_{AML}, respectively, and in related corresponding entities.

5.2. Synthetic image tests

Repeated experiments were performed in order to collect results of statistical significance. A set of 3D points was synthesised in a cuboid of dimensions $3\,\mathrm{m} \times 1.5\,\mathrm{m} \times 3\,\mathrm{m}$ with 5 points equally spaced along each direction. The images were 3000×2000 pixels in size, with square pixels of side $9\,\mu$m. The 125 world points were captured by three perspective cameras with focal length of 3600 pixels, placed at $\tilde{\mathbf{C}}_1 = [-5, 3, 1.5]^\mathsf{T}$, $\tilde{\mathbf{C}}_2 = [0, 0, 0]^\mathsf{T}$, and $\tilde{\mathbf{C}}_3 = [3, 3, 1.5]^\mathsf{T}$ to provide "true" matches. The center of the cuboid was located 5 m away from the world origin at $\tilde{\mathbf{C}}_2$. Figure 3 depicts the 3D scene with the camera positions and orientations. Each true image point was then perturbed by homogeneous Gaussian noise with zero mean and standard deviation of $\sigma = 2$ pixels. The resulting noise-contaminated triples of matching points were used as input to the five algorithms.

For each of 200 experiments, all five methods were employed to compute a tensor estimate. Table 1 below shows averages over the total number of trials. FNS failed to converge 27 times in the 200 tests and so was not included in the table. The first column considers the J_{AML} cost function and shows that the iterative schemes produce estimates which achieve very similar cost values.

The second column gives perhaps the results of a more critical test coming from using the maximum likelihood

Table 1. Average J_{AML} and J_{ML} cost values for synthetic data experiments.

Methods	J_{AML}	J_{ML}
NALS	1444.4	1444.4
HEIV	1426.8	1427.0
RFNS	1426.8	1426.8
GS	1427.0	1427.1

function, J_{ML}. For a trifocal tensor estimate $\widehat{\boldsymbol{\theta}}$ obtained by a method other than GS, $J_{\mathrm{ML}}(\widehat{\boldsymbol{\theta}})$ was calculated by minimising the reprojection error

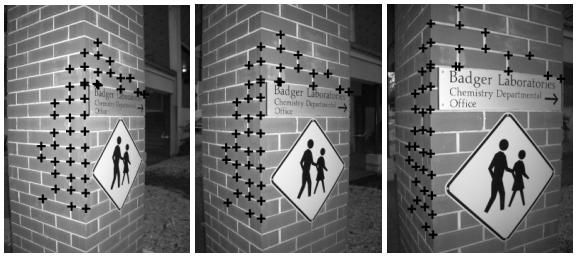
$$\sum_{i=1}^{n} \left(d(\mathbf{m}_i, \hat{\mathbf{m}}_i)^2 + d(\mathbf{m}_i', \hat{\mathbf{m}}_i')^2 + d(\mathbf{m}_i'', \hat{\mathbf{m}}_i'')^2 \right) \quad (12)$$

over all points $\hat{\mathbf{m}}_i = \mathcal{N}(\mathtt{P}\mathbf{M}_i)$, $\hat{\mathbf{m}}_i' = \mathcal{N}(\hat{\mathtt{P}}'\mathbf{M}_i)$, and $\hat{\mathbf{m}}_i'' = \mathcal{N}(\hat{\mathtt{P}}''\mathbf{M}_i)$, where $\hat{\mathtt{P}}'$ and $\hat{\mathtt{P}}''$ are retrieved from $\widehat{\boldsymbol{\theta}}$ [5] and kept fixed. Note the difference with GS where to find $\widehat{\boldsymbol{\theta}}$ minimising the reprojection error, the $\hat{\mathbf{m}}_i, \hat{\mathbf{m}}_i', \hat{\mathbf{m}}_i''$, and $\hat{\mathtt{P}}'$ and $\hat{\mathtt{P}}''$ are allowed to vary simultaneously. Here $\mathcal{N}(\mathbf{m}) = \mathbf{m}/m^3$ is a normalisation procedure whose application ensures that the third coordinate of a given planar point is unity, and $d(\mathbf{m}, \mathbf{n})$ denotes the Euclidean distance between the planar points $\mathbf{m}$ and $\mathbf{n}$ that have been normalised in the above sense. The $\mathbf{M}_i$ are initially obtained by triangulating from the $\mathbf{m}_i, \mathbf{m}_i'$, and $\mathbf{m}_i''$, and are then recomputed in each optimisation step of an iterative scheme (typically, and in our case, the Levenberg-Marquardt algorithm) that minimises the reprojection error. Upon inspection, we find that RFNS estimates produce very competitive cost function values in comparison with GS estimates, and RFNS achieves the same accuracy as HEIV, which matches expectations.

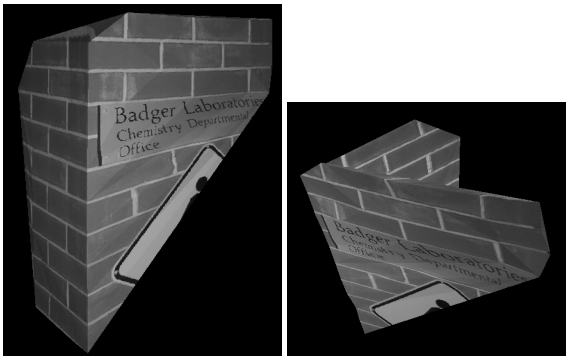
5.3. Real image test

Three images were acquired by a hand-held camera and registered using a trifocal tensor computed with RFNS to then build a 3D model, see Figure 4. To ensure convergence, a modification of RFNS in step 3 became necessary. With $\mathbf{v}_{i,k}$ the normalised eigenvector corresponding to the ith smallest eigenvalue of $\mathtt{X}'_{\boldsymbol{\mu}_{k-1}}$, the update $\boldsymbol{\mu}_k$ was defined as the result of normalising $\sum_{i=1}^{3} (\boldsymbol{\mu}_{k-1}^\mathsf{T} \mathbf{v}_{i,k}) \mathbf{v}_{i,k}$.

Table 2 shows the results of applying the previous five algorithms to the image data points. Note that HEIV, RFNS and GS give the best results and are essentially inseparable. FNS failed to converge to a sensible tensor estimate.



(a) Original



(b) Reconstruction

Figure 4. Chemistry Department. (a) Each image is 600×800 pixels in size with 44 putative points identified. (b) 3D model obtained from RFNS trifocal tensor estimate.

Table 2. J_{AML} and J_{ML} cost values for real data experiment.

Methods	J_{AML}	J_{ML}
NALS	57.1	57.1
HEIV	37.3	37.2
RFNS	37.2	37.2
GS	37.2	37.2

6. Conclusion

A novel parameter estimation method, RFNS, was proposed for problems in which the relationship between parameters and image data is expressed as a system of equations. The original FNS method operates over the entire parameter space, whereas the newly proposed method operates on a subspace of lower dimension and recuperates the missing degrees of freedom in a single final step. The performance of RFNS was demonstrated on the problem of trifocal tensor estimation. It has been shown that RFNS is more reliable than the original version of the method, often capable of generating a stable solution when FNS fails to do so. When compared to GS, RFNS gives almost identical results, both in terms of the J_{AML} residual and GS's ML cost function residual.

References

[1] W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley. On the fitting of surfaces to data with covariances. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1294–1303, 2000.

[2] W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley. From FNS to HEIV: A link between two vision parameter estimation methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(2):264–268, 2004.

[3] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*. Kluwer, Dordrecht, 1996.

[4] O. Faugeras, Q.-T. Luong, and T. Papadopoulo. *The Geometry of Multiple Images*. MIT Press, Cambridge, Mass., 2001.

[5] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[6] R. I. Hartley. Lines and points in three views and the trifocal tensor. *Int. J. Comput. Vision*, 22(2):125–140, 1997.

[7] K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice*. Elsevier, Amsterdam, 1996.

[8] Y. Leedan and P. Meer. Heteroscedastic regression in computer vision: problems with bilinear constraint. *Int. J. Comput. Vision*, 37(2):127–150, 2000.

[9] H. Lütkepol. *Handbook of Matrices*. John Wiley & Sons, Chichester, 1996.

[10] http://www.caip.rutgers.edu/riul/research/code.html.

[11] B. Matei and P. Meer. A versatile method for trifocal tensor estimation. In *Proc. Eighth Int. Conf. Computer Vision*, volume 2, pages 578–585, 2001.

[12] B. Matei and P. Meer. Estimation of nonlinear errors-in-variables models for computer vision applications. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(10):1537–1552, 2006.

[13] T. Scoleri, W. Chojnacki, and M. J. Brooks. A multi-objective parameter estimator for image mosaicing. In *Proc. IEEE Int. Symposium Signal Processing and its Applications*, pages 551–554, 2005.

[14] A. Shashua. Trilinear tensor: The fundamental construct of multiple-view geometry and its applications. In *Proc. Int. Workshop Algebraic Frames for The Perception-Action Cycle*, volume 1315 of *Lecture Notes in Computer Science*, pages 190–206. Springer, 1997.

[15] P. H. S. Torr and A. Zisserman. Robust parametrization and computation of the trifocal tensor. *Image and Vision Computing*, 15:591–605, 1997.

[16] I. Zoghlami, O. Faugeras, and R. Deriche. Using geometric corners to build a 2D mosaic from a set of images. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 420–425, 1997.