Proc. VIIth Digital Image Computing: Techniques and Applications, Sun C., Talbot H., Ourselin S. and Adriaansen T. (Eds.), 10-12 Dec. 2003, Sydney

FNS, CFNS and HEIV: Extending Three Vision Parameter Estimation Methods

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

School of Computer Science, Adelaide University, SA 5005, Australia
{wojtek,mjb,hengel,dg}@cs.adelaide.edu.au

Abstract. Estimation of parameters from image tokens is a central problem in computer vision. FNS, CFNS and HEIV are three recently developed methods for solving special but important cases of this problem. The schemes are means for finding unconstrained (FNS, HEIV) and constrained (CFNS) minimisers of cost functions. In earlier work of the authors, FNS, CFNS and a version of HEIV were applied to a specific cost function. Here we outline an extension of the approach to more general cost functions. This allows the FNS, CFNS and HEIV methods to be placed within a common framework.

1 Introduction

A common task in computer vision is the estimation of the parameters that describe a relationship between image feature locations. The estimation problem can often be reduced to minimising a cost function. FNS, CFNS and HEIV are three recently developed techniques for finding minimisers of cost functions underpinning a special but important class of estimation problems. FNS and HEIV aim to determine unconstrained minimisers, while CFNS seeks to isolate constrained minimisers. In earlier work of the authors [2, 4, 3], FNS, CFNS and a core version of HEIV were applied to a specific cost function. The purpose of this article is to outline how the methods can be extended to cope with more general cost functions, including the cost function that pertains to the original version of HEIV [8] which is different from the core version.

We start by introducing an optimal, maximum likelihood cost function that is appropriate for a class of estimation problems. We then evolve two approximations to this function. One of these is the cost function to which the standard versions of FNS and CFNS and the core version of HEIV apply. The other is the cost function recognised here as the function underlying the original version of HEIV. Both of these functions have a similar form and can be viewed as specialisations of a single model function. The subsequent development, based largely on a critical review of our earlier work, is concerned with advancing variants of FNS, HEIV and CFNS for this model function. Finally, we discuss the effects of applying the derived methods to the two approximated maximum likelihood functions.

2 Estimation Problem

Relationships between image tokens can often be arranged into parametric models. Of particular importance are models expressed by means of a *principal constraint* of the form

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

Here $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_l]^T$ is a vector that represents parameters describing a particular model; $\boldsymbol{x} = [x_1, \ldots, x_k]^T$ is a vector that represents an ideal data point conforming to the model; and $\boldsymbol{u}(\boldsymbol{x}) = [u_1(\boldsymbol{x}), \ldots, u_l(\boldsymbol{x})]^T$ is a vector with the ideal data point transformed so that: (i) each component $u_i(\boldsymbol{x})$ is a quadratic form in the compound vector $[\boldsymbol{x}^T, 1]^T$, (ii) the last component $u_l(\boldsymbol{x})$ is equal to 1. In some cases, the parameters are subject to an *ancillary constraint* not involving model points. A common form of the ancillary constraint is

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

where, for some real number κ, ϕ is a scalar-valued function *homogeneous* of degree κ—that is such that $\phi(t\boldsymbol{\theta}) = t^\kappa \phi(\boldsymbol{\theta})$ for every non-zero scalar t.

Associated with (1) and (2) is the following estimation problem: Given a collection $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ of *observed* data points and a meaningful *cost function* that characterises the extent to which any particular $\boldsymbol{\theta}$ fails to satisfy the system of copies of equation (1) associated with $\boldsymbol{x} = \boldsymbol{x}_i$ $(i = 1, \ldots, n)$, find $\boldsymbol{\theta} \neq \boldsymbol{0}$ satisfying (2) for which the cost function attains its minimum.

Example estimation problems of the above form include the estimation of the coefficients of the *epipolar equation* [5] and the *differential epipolar equation* [1], and *conic fitting* [6]. Each of the first two problems involves a separate ancillary *cubic* constraint, while the last problem involves no constraint.

3 The ML Cost Function

A statistically viable cost function can be derived by adopting the measurement model whereby the observed data points are generated from model points through a Gaussian error process. For each $i = 1, \ldots, n$, let $\boldsymbol{\Lambda}_{\boldsymbol{x}_i}$ be a $k \times k$ symmetric *covariance matrix* quantifying errors in the measurement of the data point $\boldsymbol{x}_i$. The measurement model combined with the principle of *maximum likelihood* produces an adequate discrepancy measure in the form of the squared *Mahalanobis distance*

$$d^2_{\mathrm{Mahal}}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n; \overline{\boldsymbol{x}}_1, \ldots, \overline{\boldsymbol{x}}_n) = \sum_{i=1}^{n} (\boldsymbol{x}_i - \overline{\boldsymbol{x}}_i)^T \boldsymbol{\Lambda}_{\boldsymbol{x}_i}^{-1} (\boldsymbol{x}_i - \overline{\boldsymbol{x}}_i)$$

between the data points $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ and the model points $(\overline{\boldsymbol{x}}_1, \ldots, \overline{\boldsymbol{x}}_n)$. For each $\boldsymbol{\theta} \neq \boldsymbol{0}$, when restricted to the set of those $(\overline{\boldsymbol{x}}_1, \ldots, \overline{\boldsymbol{x}}_n)$ that satisfy

$$\boldsymbol{\theta}^T \boldsymbol{u}(\overline{\boldsymbol{x}}_1) = \cdots = \boldsymbol{\theta}^T \boldsymbol{u}(\overline{\boldsymbol{x}}_n) = 0,$$

the function $(\overline{\boldsymbol{x}}_1, \ldots, \overline{\boldsymbol{x}}_n) \mapsto d^2_{\mathrm{Mahal}}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n; \overline{\boldsymbol{x}}_1, \ldots, \overline{\boldsymbol{x}}_n)$ attains a constrained minimum at some point $(\overline{\boldsymbol{x}}_1^{\boldsymbol{\theta}}, \ldots, \overline{\boldsymbol{x}}_n^{\boldsymbol{\theta}})$. All these minima can be assembled into a function by setting

$$J_{\mathrm{ML}}(\boldsymbol{\theta}) = d^2_{\mathrm{Mahal}}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n; \overline{\boldsymbol{x}}_1^{\boldsymbol{\theta}}, \ldots, \overline{\boldsymbol{x}}_n^{\boldsymbol{\theta}}).$$

This function is the optimal, maximum likelihood cost function for $\boldsymbol{\theta}$-estimation. The minimiser of J_{ML}, $\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}$, is the maximum likelihood estimate of $\boldsymbol{\theta}$. Of all candidate parameter vectors, $\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}$ is the preferred vector that makes the observed data as likely as possible.

Finding $(\overline{\boldsymbol{x}}_1^{\boldsymbol{\theta}}, \ldots, \overline{\boldsymbol{x}}_n^{\boldsymbol{\theta}})$ for each $\boldsymbol{\theta}$ is a daunting task, and so direct minimisation of J_{ML} is rather impractical. A more feasible approach is to seek to minimise an appropriate approximation of J_{ML} that captures near-optimality. A key to the development of various approximations is an alternative formula for J_{ML}.

For each $\gamma = 1, \ldots, l$, the component $u_\gamma(\boldsymbol{x})$ is a quadratic function in $\boldsymbol{x}$. Therefore $\partial^2_{\boldsymbol{x}\boldsymbol{x}} u_\gamma(\boldsymbol{y}) = [(\partial^2 u_\gamma / \partial x_i \partial x_j)(\boldsymbol{y})]_{1 \leq i,j \leq k}$, the Hessian matrix of u_γ at $\boldsymbol{y}$, is independent of $\boldsymbol{y}$. Denote by $\boldsymbol{H}_\gamma$ the unique value of $\partial^2_{\boldsymbol{x}\boldsymbol{x}} u_\gamma$. Let

$$\mu_\gamma(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{y})^T \boldsymbol{H}_\gamma (\boldsymbol{x} - \boldsymbol{y}) \quad (1 \leq \gamma \leq l)$$

and $\boldsymbol{\mu}(\boldsymbol{x}, \boldsymbol{y}) = [\mu_1(\boldsymbol{x}, \boldsymbol{y}), \ldots, \mu_l(\boldsymbol{x}, \boldsymbol{y})]^T$. Applying the method of Lagrange Multipliers to the constrained minimiser $(\overline{\boldsymbol{x}}_1^{\boldsymbol{\theta}}, \ldots, \overline{\boldsymbol{x}}_n^{\boldsymbol{\theta}})$, one can establish the following crucial formula:

$$J_{\mathrm{ML}}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \frac{(\boldsymbol{\theta}^T(\boldsymbol{u}(\boldsymbol{x}_i) - \boldsymbol{\mu}(\boldsymbol{x}_i, \overline{\boldsymbol{x}}_i^{\boldsymbol{\theta}})))^2}{\boldsymbol{\theta}^T \partial_{\boldsymbol{x}} \boldsymbol{u}(\overline{\boldsymbol{x}}_i^{\boldsymbol{\theta}}) \boldsymbol{\Lambda}_{\boldsymbol{x}_i} \partial_{\boldsymbol{x}} \boldsymbol{u}(\overline{\boldsymbol{x}}_i^{\boldsymbol{\theta}})^T \boldsymbol{\theta}}. \tag{3}$$

Here $\partial_{\boldsymbol{x}} \boldsymbol{u}(\boldsymbol{y}) = [(\partial u_i / \partial x_j)(\boldsymbol{y})]_{1 \leq i \leq l, 1 \leq j \leq k}$ denotes the Jacobian matrix of $\boldsymbol{u}$ at $\boldsymbol{y}$.

4 Two AML Cost Functions

Eq. (3) can be exploited to derive two approximations to J_{ML}. In both of them $\partial_{\boldsymbol{x}} \boldsymbol{u}(\overline{\boldsymbol{x}}_i^{\boldsymbol{\theta}})$ will be replaced by $\partial_{\boldsymbol{x}} \boldsymbol{u}(\boldsymbol{x}_i)$. In addition, one approximation will treat $\boldsymbol{\mu}_i(\boldsymbol{x}_i, \overline{\boldsymbol{x}}_i^{\boldsymbol{\theta}})$ as an irrelevant second-order term and set it to zero. The other approximation will replace $\boldsymbol{\mu}_i(\boldsymbol{x}_i, \overline{\boldsymbol{x}}_i^{\boldsymbol{\theta}})$ by an average value of some kind. Careful analysis shows that in the latter case a natural replacement for $\boldsymbol{\mu}(\boldsymbol{x}_i, \overline{\boldsymbol{x}}_i^{\boldsymbol{\theta}})$ is the ith *second-order correction* $\boldsymbol{\mu}(\boldsymbol{x}_i) = [\mu_1(\boldsymbol{x}_i), \ldots, \mu_l(\boldsymbol{x}_i)]^T$ defined by

$$\mu_\gamma(\boldsymbol{x}) = \frac{1}{2} \operatorname{tr}(\boldsymbol{H}_\gamma \boldsymbol{\Lambda}_{\boldsymbol{x}}) \quad (1 \leq \gamma \leq l).$$

Note that the last component of $\boldsymbol{\mu}(\boldsymbol{x})$, $\mu_l(\boldsymbol{x})$, is null, since $u_l(\boldsymbol{x}) = 1$ and, consequently, $\boldsymbol{H}_l = \boldsymbol{0}$. Thus the first approximation of J_{ML} takes the form

$$J^{(1)}_{\mathrm{AML}}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \frac{(\boldsymbol{\theta}^T \boldsymbol{u}(\boldsymbol{x}_i))^2}{\boldsymbol{\theta}^T \partial_{\boldsymbol{x}} \boldsymbol{u}(\boldsymbol{x}_i) \boldsymbol{\Lambda}_{\boldsymbol{x}_i} \partial_{\boldsymbol{x}} \boldsymbol{u}(\boldsymbol{x}_i)^T \boldsymbol{\theta}},$$

whereas the second approximation is given by

$$J_{\mathrm{AML}}^{(2)}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \frac{(\boldsymbol{\theta}^T(\boldsymbol{u}(\boldsymbol{x}_i) - \boldsymbol{\mu}_i))^2}{\boldsymbol{\theta}^T \partial_{\boldsymbol{x}} \boldsymbol{u}(\boldsymbol{x}_i) \boldsymbol{\Lambda}_{\boldsymbol{x}_i} \partial_{\boldsymbol{x}} \boldsymbol{u}(\boldsymbol{x}_i)^T \boldsymbol{\theta}}.$$

For each $i = 1, \ldots, n$, let $\boldsymbol{v}_1(\boldsymbol{x}_i) = \boldsymbol{u}(\boldsymbol{x}_i)$, $\boldsymbol{v}_2(\boldsymbol{x}_i) = \boldsymbol{u}(\boldsymbol{x}_i) - \boldsymbol{\mu}(\boldsymbol{x}_i)$, $\boldsymbol{A}_i^{(\alpha)} = \boldsymbol{v}_\alpha(\boldsymbol{x}_i)\boldsymbol{v}_\alpha(\boldsymbol{x}_i)^T$ $(\alpha = 1, 2)$, and $\boldsymbol{B}_i = \partial_{\boldsymbol{x}} \boldsymbol{u}(\boldsymbol{x}_i) \boldsymbol{\Lambda}_{\boldsymbol{x}_i} \partial_{\boldsymbol{x}} \boldsymbol{u}(\boldsymbol{x}_i)^T$. With this notation, $J_{\mathrm{AML}}^{(\alpha)}$ can be simply written as

$$J_{\mathrm{AML}}^{(\alpha)}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \frac{\boldsymbol{\theta}^T \boldsymbol{A}_i^{(\alpha)} \boldsymbol{\theta}}{\boldsymbol{\theta}^T \boldsymbol{B}_i \boldsymbol{\theta}} \quad (\alpha = 1, 2).$$

Note that, since $u_l(\boldsymbol{x}) = 1$ and $\mu_l(\boldsymbol{x}) = 0$, the last component of $\boldsymbol{v}_\alpha(\boldsymbol{x})$, $v_{\alpha,l}(\boldsymbol{x})$, equals 1.

5 Model Cost Function

The functions $J_{\mathrm{AML}}^{(1)}$ and $J_{\mathrm{AML}}^{(2)}$ have a similar structure and can be subsumed into a single model function

$$J_{\mathrm{AML}}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \frac{\boldsymbol{\theta}^T \boldsymbol{A}_i \boldsymbol{\theta}}{\boldsymbol{\theta}^T \boldsymbol{B}_i \boldsymbol{\theta}},$$

where all the $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ are non-negative definite $l \times l$ matrices. By convention, J_{AML} will be referred to as the *approximated maximum likelihood* cost function. The unconstrained minimiser of J_{AML} will be denoted $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u$, and if an ancillary constraint (as per (2)) applies, the constrained minimiser of J_{AML} will be denoted $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}$. We shall mainly consider J_{AML} with the $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ such that

$$\boldsymbol{A}_i = \boldsymbol{v}(\boldsymbol{x}_i)\boldsymbol{v}(\boldsymbol{x}_i)^T \tag{4}$$

$$\boldsymbol{B}_i = \partial_{\boldsymbol{x}} \boldsymbol{v}(\boldsymbol{x}_i) \boldsymbol{\Lambda}_{\boldsymbol{x}_i} \partial_{\boldsymbol{x}} \boldsymbol{v}(\boldsymbol{x}_i)^T \tag{5}$$

for some $\boldsymbol{v}(\boldsymbol{x}) = [v_1(\boldsymbol{x}), \ldots, v_l(\boldsymbol{x})]^T$ with $v_l(\boldsymbol{x}) = 1$. Note that with $\boldsymbol{v}(\boldsymbol{x}) = \boldsymbol{v}_\alpha(\boldsymbol{x})$, $J_{\mathrm{AML}}^{(\alpha)}$ recovers J_{AML}.

6 Variational Equation

The unconstrained minimiser $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u$ satisfies the *variational equation*

$$[\partial_{\boldsymbol{\theta}} J_{\mathrm{AML}}(\boldsymbol{\theta})]_{\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u} = \boldsymbol{0}^T \tag{6}$$

with $\partial_{\boldsymbol{\theta}} J_{\mathrm{AML}}$ the row vector of the partial derivatives of J_{AML} with respect to $\boldsymbol{\theta}$. It is readily verified that

$$[\partial_{\boldsymbol{\theta}} J_{\mathrm{AML}}(\boldsymbol{\theta})]^T = 2\boldsymbol{X}_{\boldsymbol{\theta}} \boldsymbol{\theta}, \tag{7}$$

where $\boldsymbol{X}_{\boldsymbol{\theta}}$ is an $l \times l$ symmetric matrix given by

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

Thus (6) can be rewritten as

$$[\boldsymbol{X}_{\boldsymbol{\theta}} \boldsymbol{\theta}]_{\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}}} = \mathbf{0}. \tag{8}$$

The latter equation provides the basis for isolating $\widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}}$.

There are two fundamental methods for solving (8). One is the *fundamental numerical scheme* (FNS) introduced by Chojnacki et al. [2]. Another is the *heteroscedastic errors-in-variables* (HEIV) *scheme* that was first proposed by Leedan and Meer [8] and further developed by Matei and Meer [10, 9].

7 Fundamental Numerical Scheme

A vector $\boldsymbol{\theta}$ satisfies (8) if and only if it is a solution of the *ordinary* eigenvalue problem

$$\boldsymbol{X}_{\boldsymbol{\theta}} \boldsymbol{\xi} = \lambda \boldsymbol{\xi} \tag{9}$$

corresponding to the eigenvalue $\lambda = 0$. This suggests an iterative method for solving (8) whereby if $\boldsymbol{\theta}_c$ is a current approximate solution, then an updated solution $\boldsymbol{\theta}_+$ is a vector chosen from that eigenspace of $\boldsymbol{X}_{\boldsymbol{\theta}_c}$ which most closely approximates the null space of $\boldsymbol{X}_{\boldsymbol{\theta}}$; this eigenspace is, of course, the one corresponding to the eigenvalue closest to zero in absolute value. The process can be started by computing the *algebraic least squares* (ALS) *estimate*, $\widehat{\boldsymbol{\theta}}_{\mathrm{ALS}}$, defined as the unconstrained minimiser of the cost function $J_{\mathrm{ALS}}(\boldsymbol{\theta}) = \|\boldsymbol{\theta}\|^{-2} \sum_{i=1}^{n} \boldsymbol{\theta}^T \boldsymbol{A}_i \boldsymbol{\theta}$, with $\|\boldsymbol{\theta}\| = (\sum_{j=1}^{l} \theta_j^2)^{1/2}$. The estimate $\widehat{\boldsymbol{\theta}}_{\mathrm{ALS}}$ coincides, up to scale, with an eigenvector of $\sum_{i=1}^{n} \boldsymbol{A}_i$ associated with the smallest eigenvalue. When the $\boldsymbol{A}_i$ satisfy (4), this eigenvector can be found by performing *singular-value decomposition* on the matrix $[\boldsymbol{v}(\boldsymbol{x}_1), \ldots, \boldsymbol{v}(\boldsymbol{x}_i)]^T$. The overall procedure is summarised in Algorithm 1.

Algorithm 1. Fundamental numerical scheme

1. Set $\boldsymbol{\theta}$ to $\widehat{\boldsymbol{\theta}}_{\mathrm{ALS}}$.
2. Repeat:
   (a) Compute the matrix $\boldsymbol{X}_{\boldsymbol{\theta}}$;
   (b) Compute a normalised eigenvector of $\boldsymbol{X}_{\boldsymbol{\theta}}$ corresponding to the eigenvalue closest to zero (in absolute value);
   (c) Take the computed eigenvector for an update of $\boldsymbol{\theta}$;
   until convergence.

8 HEIV: A Basic Form

Given the representation $\boldsymbol{X}_{\boldsymbol{\theta}} = \boldsymbol{M}_{\boldsymbol{\theta}} - \boldsymbol{N}_{\boldsymbol{\theta}}$, where $\boldsymbol{M}_{\boldsymbol{\theta}} = \sum_{i=1}^{n} (\boldsymbol{\theta}^T \boldsymbol{B}_i \boldsymbol{\theta})^{-1} \boldsymbol{A}_i$ and $\boldsymbol{N}_{\boldsymbol{\theta}} = \sum_{i=1}^{n} (\boldsymbol{\theta}^T \boldsymbol{A}_i \boldsymbol{\theta})(\boldsymbol{\theta}^T \boldsymbol{B}_i \boldsymbol{\theta})^{-2} \boldsymbol{B}_i$, the variational equation (8) can be restated as

$$\boldsymbol{M}_{\boldsymbol{\theta}} \boldsymbol{\theta} = \boldsymbol{N}_{\boldsymbol{\theta}} \boldsymbol{\theta}, \tag{10}$$

where the evaluation at $\widehat{\boldsymbol{\theta}}^{u}_{\mathrm{AML}}$ is dropped for clarity. The matrices $\boldsymbol{M}_{\boldsymbol{\theta}}$ and $\boldsymbol{N}_{\boldsymbol{\theta}}$ are non-negative definite (with $\boldsymbol{M}_{\boldsymbol{\theta}}$, a sum of n summands, generically positive definite if $n \geq l$), so $\boldsymbol{\theta}$ can be viewed as a solution of the *generalised* eigenvalue problem

$$\boldsymbol{M}_{\boldsymbol{\theta}} \boldsymbol{\xi} = \lambda \boldsymbol{N}_{\boldsymbol{\theta}} \boldsymbol{\xi} \tag{11}$$

corresponding to the eigenvalue $\lambda = 1$. The heteroscedastic errors-in-variables scheme in *basic* form, or HEIV *with intercept* [9], exploits the above eigenvalue problem in a manner analogous to that in which FNS utilises the eigenvalue problem (9). The details are given in Algorithm 2.

Algorithm 2. Basic HEIV scheme

1. Set $\boldsymbol{\theta}$ to $\widehat{\boldsymbol{\theta}}_{\mathrm{ALS}}$.
2. Repeat:
   (a) Compute the matrices $\boldsymbol{M}_{\boldsymbol{\theta}}$ and $\boldsymbol{N}_{\boldsymbol{\theta}}$;
   (b) Compute a normalised eigenvector of the eigenvalue problem

$$\boldsymbol{M}_{\boldsymbol{\theta}} \boldsymbol{\xi} = \lambda \boldsymbol{N}_{\boldsymbol{\theta}} \boldsymbol{\xi}$$

   corresponding to the eigenvalue closest to 1;
   (c) Take the computed eigenvector for an update of $\boldsymbol{\theta}$;
   until convergence.

9 Reduced Variational Equation

If the $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ satisfy (4) and (5), respectively, for some $\boldsymbol{v}(\boldsymbol{x}) = [\boldsymbol{z}(\boldsymbol{x})^T, 1]^T$, where $\boldsymbol{z}(\boldsymbol{x})$ is a vector of length $l-1$, then the variational equation can be re-expressed as a system of equations. To see how this can be done, first partition the parameter vector as $\boldsymbol{\theta} = [\boldsymbol{\eta}^T, \alpha]^T$ with $\boldsymbol{\eta}$ a length $l-1$ vector and α a scalar. Further, let $\overline{\boldsymbol{z}} = (\sum_{i=1}^{n} \beta_i)^{-1} \sum_{i=1}^{n} \beta_i \boldsymbol{z}_i$ with $\beta_i = (\boldsymbol{\eta}^T \boldsymbol{B}_i^0 \boldsymbol{\eta})^{-1}$ and $\boldsymbol{B}_i^0 = \partial_{\boldsymbol{x}} \boldsymbol{z}(\boldsymbol{x}_i) \boldsymbol{\Lambda}_{\boldsymbol{x}_i} \partial_{\boldsymbol{x}} \boldsymbol{z}(\boldsymbol{x}_i)^T$, and let $\boldsymbol{z}_i' = \boldsymbol{z}_i - \overline{\boldsymbol{z}}$ for each $i = 1, \ldots, n$. Finally, define two $(l-1) \times (l-1)$ matrices $\boldsymbol{M}'_{\boldsymbol{\eta}} = \sum_{i=1}^{n} \beta_i \boldsymbol{z}_i' \boldsymbol{z}_i'^T$ and $\boldsymbol{N}'_{\boldsymbol{\eta}} = \sum_{i=1}^{n} \left(\beta_i \boldsymbol{z}_i'^T \boldsymbol{\eta}\right)^2 \boldsymbol{B}_i^0$. A fundamental result that can now be established [3] is that $\boldsymbol{\theta} = [\boldsymbol{\eta}^T, \alpha]^T$ satisfies (10) if and only if the following system of equations holds:

$$\boldsymbol{M}'_{\boldsymbol{\eta}} \boldsymbol{\eta} = \boldsymbol{N}'_{\boldsymbol{\eta}} \boldsymbol{\eta}, \tag{12}$$

$$\alpha = -\overline{\boldsymbol{z}}^T \boldsymbol{\eta}. \tag{13}$$

The first of these equations involves only $\boldsymbol{\eta}$ and can be solved in isolation; the second expresses α in terms of $\boldsymbol{\eta}$. Of the two constraints, the first plays a leading role and is called the *reduced variational equation.* A key feature of this equation is that its right-hand side matrix $\boldsymbol{N}'_{\boldsymbol{\eta}}$ is generically *positive definite* if $n \geq l$. In contrast, $\boldsymbol{N}_{\boldsymbol{\theta}}$ is singular, since all the $\boldsymbol{B}_i$ have the length l vector $[0, \ldots, 0, 1]^T$ in their respective null spaces.

10 HEIV: A Reduced Form

Define the algebraic least squares estimates $\widehat{\boldsymbol{\eta}}_{\mathrm{ALS}}$ and $\widehat{\alpha}_{\mathrm{ALS}}$ as the respective components in the representation $\widehat{\boldsymbol{\theta}}_{\mathrm{ALS}} = [(\widehat{\boldsymbol{\eta}}_{\mathrm{ALS}})^T, \widehat{\alpha}_{\mathrm{ALS}}]^T$. Analogously, define the unconstrained approximated maximum likelihood estimates $\widehat{\boldsymbol{\eta}}^u_{\mathrm{AML}}$ and $\widehat{\alpha}^u_{\mathrm{AML}}$ via the decomposition $\widehat{\boldsymbol{\theta}}^u_{\mathrm{AML}} = [(\widehat{\boldsymbol{\eta}}^u_{\mathrm{AML}})^T, \widehat{\alpha}^u_{\mathrm{AML}}]^T$. In view of (13), $\widehat{\alpha}^u_{\mathrm{AML}}$ is uniquely determined by $\widehat{\boldsymbol{\eta}}^u_{\mathrm{AML}}$—taking $\overline{\boldsymbol{z}}$ with the $\beta_i = ((\widehat{\boldsymbol{\eta}}^u_{\mathrm{AML}})^T \boldsymbol{B}_i^0 \, \widehat{\boldsymbol{\eta}}^u_{\mathrm{AML}})^{-1}$ results in $\widehat{\alpha}^u_{\mathrm{AML}} = -\overline{\boldsymbol{z}}^T \widehat{\boldsymbol{\eta}}^u_{\mathrm{AML}}$. Now, the matrix $\boldsymbol{N}'_{\boldsymbol{\eta}}$ is generically positive-definite, and so the generalised eigenvalue problem $\boldsymbol{M}'_{\boldsymbol{\eta}}\boldsymbol{\zeta} = \lambda \boldsymbol{N}'_{\boldsymbol{\eta}}\boldsymbol{\zeta}$ is non-degenerate. Accordingly, $\widehat{\boldsymbol{\eta}}^u_{\mathrm{AML}}$ can be determined via a simple modification of the HEIV algorithm. The steps of this HEIV scheme in *reduced* form, or *HEIV without intercept* [9], are given in Algorithm 3.

Algorithm 3. Reduced HEIV scheme

1. Set $\boldsymbol{\eta}$ to $\widehat{\boldsymbol{\eta}}_{\mathrm{ALS}}$.
2. Repeat:
   (a) Compute the matrices $\boldsymbol{M}'_{\eta}$ and $\boldsymbol{N}'_{\eta}$;
   (b) Compute a normalised eigenvector of the eigenvalue problem

$$\boldsymbol{M}'_{\eta}\boldsymbol{\zeta} = \lambda \boldsymbol{N}'_{\eta}\boldsymbol{\zeta}$$

   corresponding to the eigenvalue closest to 1 and take this eigenvector for $\boldsymbol{\eta}$;
   (c) Take the computed eigenvector for an update of $\boldsymbol{\eta}$;
   until convergence.

11 Variational System

Applied to the constrained minimiser $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}$, the method of Lagrange Multipliers yields

$$\begin{aligned} [\partial_{\boldsymbol{\theta}} J_{\mathrm{AML}}(\boldsymbol{\theta}) + \lambda \partial_{\boldsymbol{\theta}} \phi(\boldsymbol{\theta})]_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}} &= \mathbf{0}^T, \\ \phi(\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}) &= 0, \end{aligned}$$

where λ is scalar. When properly combined with the identity $\partial_{\boldsymbol{\theta}}\phi(\boldsymbol{\theta})\boldsymbol{\theta} = \kappa\phi(\boldsymbol{\theta})$ obtained by differentiating (2) with respect to t and evaluating at $t = 1$, this *variational system* can be converted into a single equation similar to (8). Of many equivalent forms, the one useful to us reads

$$[\boldsymbol{Q}_{\boldsymbol{\theta}}\boldsymbol{\theta}]_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}} = \mathbf{0}, \tag{14}$$

where $\boldsymbol{Q}_{\boldsymbol{\theta}} = \boldsymbol{Z}_{\boldsymbol{\theta}}^T \boldsymbol{Z}_{\boldsymbol{\theta}}$ and $\boldsymbol{Z}_{\boldsymbol{\theta}}$ is an $l \times l$ matrix defined as follows. Let $\boldsymbol{P}_{\boldsymbol{\theta}} = \boldsymbol{I}_l - \|\boldsymbol{a}_{\boldsymbol{\theta}}\|^{-2}\boldsymbol{a}_{\boldsymbol{\theta}}\boldsymbol{a}_{\boldsymbol{\theta}}^T$, where $\boldsymbol{I}_l$ denotes the $l \times l$ identity matrix and $\boldsymbol{a}_{\boldsymbol{\theta}} = [\partial_{\boldsymbol{\theta}}\phi(\boldsymbol{\theta})]^T/2$. Denote by $\boldsymbol{H}_{\boldsymbol{\theta}}$ the Hessian of J_{AML} at $\boldsymbol{\theta}$; more explicitly, $\boldsymbol{H}_{\boldsymbol{\theta}} = 2(\boldsymbol{X}_{\boldsymbol{\theta}} - \boldsymbol{T}_{\boldsymbol{\theta}})$, where

$$\boldsymbol{T}_{\boldsymbol{\theta}} = \sum_{i=1}^{n} \frac{2}{(\boldsymbol{\theta}^T\boldsymbol{B}_i\boldsymbol{\theta})^2}\Big[\boldsymbol{A}_i\boldsymbol{\theta}\boldsymbol{\theta}^T\boldsymbol{B}_i + \boldsymbol{B}_i\boldsymbol{\theta}\boldsymbol{\theta}^T\boldsymbol{A}_i - 2\frac{\boldsymbol{\theta}^T\boldsymbol{A}_i\boldsymbol{\theta}}{\boldsymbol{\theta}^T\boldsymbol{B}_i\boldsymbol{\theta}}\boldsymbol{B}_i\boldsymbol{\theta}\boldsymbol{\theta}^T\boldsymbol{B}_i\Big].$$

Let $\boldsymbol{\Phi}_{\boldsymbol{\theta}}$ be the Hessian of ϕ at $\boldsymbol{\theta}$. For each $i \in \{1, \dots, l\}$, let $\boldsymbol{e}_i$ be the length l vector whose ith entry is unital and all other entries are zero. With all the preparations now completed, we let $\boldsymbol{Z}_{\boldsymbol{\theta}} = \boldsymbol{A}_{\boldsymbol{\theta}} + \boldsymbol{B}_{\boldsymbol{\theta}} + \boldsymbol{C}_{\boldsymbol{\theta}}$, where

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

$$\boldsymbol{B}_{\boldsymbol{\theta}} = \|\boldsymbol{\theta}\|^2\|\boldsymbol{a}_{\boldsymbol{\theta}}\|^{-2}\Big[\sum_{i=1}^{l}(\boldsymbol{\Phi}_{\boldsymbol{\theta}}\boldsymbol{e}_i\boldsymbol{a}_{\boldsymbol{\theta}}^T + \boldsymbol{a}_{\boldsymbol{\theta}}\boldsymbol{e}_i^T\boldsymbol{\Phi}_{\boldsymbol{\theta}})\boldsymbol{X}_{\boldsymbol{\theta}}\boldsymbol{\theta}\boldsymbol{e}_i^T - 2\|\boldsymbol{a}_{\boldsymbol{\theta}}\|^{-2}\boldsymbol{a}_{\boldsymbol{\theta}}\boldsymbol{a}_{\boldsymbol{\theta}}^T\boldsymbol{X}_{\boldsymbol{\theta}}\boldsymbol{\theta}\boldsymbol{a}_{\boldsymbol{\theta}}^T\boldsymbol{\Phi}_{\boldsymbol{\theta}}\Big],$$

$$\boldsymbol{C}_{\boldsymbol{\theta}} = \|\boldsymbol{a}_{\boldsymbol{\theta}}\|^{-2}\kappa\Big[\frac{\phi(\boldsymbol{\theta})}{4}\boldsymbol{\Phi}_{\boldsymbol{\theta}} + \boldsymbol{a}_{\boldsymbol{\theta}}\boldsymbol{a}_{\boldsymbol{\theta}}^T - \frac{\phi(\boldsymbol{\theta})}{2}\|\boldsymbol{a}_{\boldsymbol{\theta}}\|^{-2}\boldsymbol{a}_{\boldsymbol{\theta}}\boldsymbol{a}_{\boldsymbol{\theta}}^T\boldsymbol{\Phi}_{\boldsymbol{\theta}}\Big].$$

Here, individually, the matrices $\boldsymbol{A}_{\boldsymbol{\theta}}$, $\boldsymbol{B}_{\boldsymbol{\theta}}$ and $\boldsymbol{C}_{\boldsymbol{\theta}}$ do not have any special significance and serve only to split the otherwise lengthy formula.

12 Constrained Fundamental Numerical Scheme

Letting $\boldsymbol{Q}_{\boldsymbol{\theta}}$ play the role of $\boldsymbol{X}_{\boldsymbol{\theta}}$, one can advance an algorithm fully analogous to FNS [4]. The steps of the resulting *constrained fundamental numerical scheme* (CFNS) are given in Algorithm 4.

Algorithm 4. Constrained fundamental numerical scheme

1. Set $\boldsymbol{\theta}$ to $\widehat{\boldsymbol{\theta}}_{\mathrm{ALS}}$.
2. Repeat:
   (a) Compute the matrix $\boldsymbol{Q}_{\boldsymbol{\theta}}$;
   (b) Compute a normalised eigenvector of $\boldsymbol{Q}_{\boldsymbol{\theta}}$ corresponding to the eigenvalue closest to zero (in absolute value);
   (c) Take the computed eigenvector for an update of $\boldsymbol{\theta}$;
   until convergence.

For CFNS to converge to a vector $\boldsymbol{\theta}^*$ solving (14), the zero eigenvalue of $\boldsymbol{Q}_{\boldsymbol{\theta}^*}$ must be simple, i.e., the null space of $\boldsymbol{Q}_{\boldsymbol{\theta}^*}$ must be one-dimensional, with all members being scalar multiples of $\boldsymbol{\theta}^*$. When this condition is satisfied, the algorithm seeded with an estimate close enough to $\boldsymbol{\theta}^*$ produces updates quickly converging to $\boldsymbol{\theta}^*$. In practice it is required that, for each iterate $\boldsymbol{\theta}_c$, the smallest (non-negative) eigenvalue of $\boldsymbol{Q}_{\boldsymbol{\theta}_c}$ should be sufficiently well separated from the remaining eigenvalues. Sometimes, to meet the condition, the data will have to be first suitably transformed and their covariances propagated; upon application of CNFS, the estimate will then have to be conformally readjusted (transformed back) to account for the data-cum-covariances transformation. Such is the case for fundamental matrix estimation, where an initial transformation of raw data and their covariances is necessary for a successful application of CFNS [12].

Interestingly, many other, often simpler, equivalent forms of (14) like

$$[\boldsymbol{Y}_{\boldsymbol{\theta}}\boldsymbol{\theta}]_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}} = \boldsymbol{0} \quad \text{with} \quad \boldsymbol{Y}_{\boldsymbol{\theta}} = \|\boldsymbol{\theta}\|^2 \boldsymbol{P}_{\boldsymbol{\theta}} \boldsymbol{X}_{\boldsymbol{\theta}} \boldsymbol{P}_{\boldsymbol{\theta}} + \boldsymbol{I}_l - \boldsymbol{P}_{\boldsymbol{\theta}}$$

lead to non-converging algorithms, with divergence occurring irrespective of the distance of the initial estimate from the desired limit. This reflects the rather complicated behaviour of the function that sends a symmetric matrix to the eigenspace corresponding to the eigenvalue closest to zero.

13 Discussion and Conclusion

With the general formulation of various algorithms finally accomplished, we proceed to discuss implications for the AML cost functions.

The function $J^{(1)}_{\mathrm{AML}}$ was first proposed by Kanatani [7] as a cost function capturing *geometric fitting*. An important precursor of $J^{(1)}_{\mathrm{AML}}$ was Sampson's [11] cost function for some form of *orthogonal regression*. FNS and CFNS were introduced by the authors [2, 4] to perform unconstrained and constrained minimisation of $J^{(1)}_{\mathrm{AML}}$. Later it was recognised that the core version of HEIV that was first adopted in [9] computes—in both basic and reduced forms—the unconstrained minimiser of $J^{(1)}_{\mathrm{AML}}$ [3].

The function $J^{(2)}_{\mathrm{AML}}$ introduced here is a second-order variation of $J^{(1)}_{\mathrm{AML}}$. In some cases, like that of fundamental matrix estimation, the two functions coincide. Generally, they are different, as exemplified by the problem of estimating the coefficients of the differential epipolar equation. The significance of $J^{(2)}_{\mathrm{AML}}$ is that it allows the original version of HEIV [8] to be placed within the operational framework of FNS, CFNS and the core version of HEIV. An inspection of equations (22), (23), (24) in [8] reveals that the system of equations describing the estimate produced by the HEIV scheme is equivalent to the system comprising (12) and (13) for computing the unconstrained minimiser of $J^{(2)}_{\mathrm{AML}}$. Thus the original form of HEIV turns out to be identical with the reduced HEIV scheme for computing the unconstrained minimiser of $J^{(2)}_{\mathrm{AML}}$. Of course, this minimiser can also be recovered using FNS or the basic HEIV scheme, both based on $J^{(2)}_{\mathrm{AML}}$.

Note that the original derivation of HEIV did not utilise $J^{(2)}_{\mathrm{AML}}$—see [3] for a more detailed explanation.

On a final point, CFNS offers a simultaneous extension of the FNS and HEIV methods to the case of constrained minimisation. In particular, the $J^{(1)}_{\mathrm{AML}}$ and $J^{(2)}_{\mathrm{AML}}$-based versions of CFNS are the constrained-minimisation counterparts of the core and original versions of HEIV, respectively. The first of these is the original version of CFNS as introduced in [3]. The second emerges as a new method still to be tested and compared with other techniques.

Summarising, this paper has outlined a unifying approach to three recent estimation techniques: FNS, CFNS and HEIV. The proposed formulation allows for consistent analysis of various existing algorithms and advancement of new variants.

References

1. M. J. Brooks, W. Chojnacki, and L. Baumela. Determining the egomotion of an uncalibrated camera from instantaneous optical flow. *Journal of the Optical Society of America A*, 14(10):2670–2677, 1997.
2. W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley. On the fitting of surfaces to data with covariances. *Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1294–1303, 2000.
3. W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley. From FNS to HEIV: A link between two vision parameter estimation methods. *Transactions on Pattern Analysis and Machine Intelligence*, in press.
4. W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley. A new constrained parameter estimator for computer vision applications. *Image and Vision Computing*, in press.
5. O. D. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge, Mass., 1993.
6. A. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct least square fitting of ellipses. *Transactions on Pattern Analysis and Machine Intelligence*, 21(5):476–480, 1999.
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. *International Journal of Computer Vision*, 37(2):127–150, 2000.
9. B. Matei. *Heteroscedastic Errors-in-Variables Models in Computer Vision*. PhD thesis, Department of Electrical and Computer Engineering, Rutgers University, New Brunswick, NJ, May 2001.
10. B. Matei and P. Meer. A general method for errors-in-variables problems in computer vision. In *Proceedings,IEEE Conf. on Computer Vision and Pattern Recognition, South Carolina, June 13-15, 2000*, volume 2, pages 18–25, 2000.
11. P. D. Sampson. Fitting conic sections to 'very scattered' data: An iterative refinement of the Bookstein algorithm. *Computer Graphics and Image Processing*, 18(1):97–108, 1982.
12. A. van den Hengel, W. Chojnacki, M. J. Brooks, and D. Gawley. A new constrained parameter estimator: Experiments in fundamental matrix computation. In *Proceedings of the 13th British Machine Vision Conference, Cardiff, 2-5 September, 2002*, volume 2, pages 468–476, 2002.