

From FNS to HEIV: a link between two vision parameter estimation methods

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

Abstract—

Problems requiring accurate determination of parameters from image-based quantities arise often in computer vision. Two recent, independently developed frameworks for estimating such parameters are the FNS scheme of the authors, and the HEIV scheme of Leedan & Meer. In this paper, it is shown that the two schemes constitute intimately related but different means of numerically solving a common underlying equation characterising the minimiser. The analysis is driven by the search for a non-degenerate form of a certain generalised eigenvalue problem, and this effectively leads to a new derivation of the HEIV algorithm. This work may be seen as an extension of the authors' previous efforts to rationalise and inter-relate a spectrum of estimators, including the renormalisation method of Kanatani and the normalised eight-point method of Hartley.

Index Terms— Statistical methods, maximum likelihood, (un)constrained minimisation, fundamental matrix, epipolar equation

I. INTRODUCTION

ESTIMATION of the parameters that describe a relationship between image feature locations across multiple cameras is a central problem in computer vision. Basic examples include the stereo and motion problems of estimating coefficients of the *epipolar equation* [6] and the *differential epipolar equation* [1], and conic fitting [7]. The *principal equation* applicable in a variety of situations, 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 locations 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. In some cases, the parameters are subject to an *ancillary constraint* not involving feature locations. A common form of the ancillary constraint is

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

The estimation problem associated with (1) and (2) can 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 (1) with \mathbf{x} replaced by

\mathbf{x}_i ($i = 1, \dots, n$), find $\boldsymbol{\theta} \neq \mathbf{0}$ satisfying (2) for which the cost function attains its minimum. The *Gaussian model of errors* in data combined with the *principle of maximum likelihood* leads to 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 [9], [4], [2]). 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* minimiser of J_{AML} , defines the *approximated maximum likelihood estimate* $\hat{\boldsymbol{\theta}}_{\text{AML}}$. The *unconstrained* minimiser of J_{AML} obtained by ignoring the ancillary constraint and searching over all of the parameter space defines the *weak approximated maximum likelihood estimate*, $\hat{\boldsymbol{\theta}}_{\text{AML}}^w$. 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 both $\hat{\boldsymbol{\theta}}_{\text{AML}}$ and $\hat{\boldsymbol{\theta}}_{\text{AML}}^w$ are determined only up to scale.

Various methods are available for finding $\hat{\boldsymbol{\theta}}_{\text{AML}}^w$. One is the *heteroscedastic errors-in-variables* (HEIV) *scheme* proposed by Leedan and Meer [10] and further developed by Matei and Meer [12], [11]. Another is the *fundamental numerical scheme* (FNS) introduced by the authors [4]. The latter operates over the entire parameter space, the former operates essentially on a subspace of one dimension less and recuperates the missing dimension in a single final step. This paper aims to understand the previously unclear relationship between the two schemes. It is shown that the algorithms are two different, but intimately related, means for numerically solving one and the same equation characterising $\hat{\boldsymbol{\theta}}_{\text{AML}}^w$. In the analysis that follows, FNS is taken as a starting point, and HEIV is evolved via reduction of a certain generalised eigenvalue problem to a non-degenerate form. This approach effectively results in a new derivation of the HEIV algorithm.

Determination of $\hat{\boldsymbol{\theta}}_{\text{AML}}$ is a much more complicated task than isolation of $\hat{\boldsymbol{\theta}}_{\text{AML}}^w$. Recently, the authors proposed an integrated method for calculating $\hat{\boldsymbol{\theta}}_{\text{AML}}$ that extends the FNS technique (in the case of an ancillary constraint of a slightly restricted but ubiquitous form) [5]. This work may provide a basis for designing a similar extension to the HEIV method.

• The authors are with the Department of Computer Science, University of Adelaide, SA 5005, Australia.
• Email: {wojteck, mjb, hengel, dg}@cs.adelaide.edu.au

- 1) Set $\boldsymbol{\theta}_0 = \widehat{\boldsymbol{\theta}}_{\text{ALS}}$.
- 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.

Fig. 1. Fundamental numerical scheme.

II. FUNDAMENTAL NUMERICAL SCHEME

The unconstrained minimiser $\widehat{\boldsymbol{\theta}}_{\text{AML}}^w$ satisfies the *variational equation* for unconstrained minimisation

$$[\partial_{\boldsymbol{\theta}} J_{\text{AML}}(\boldsymbol{\theta}; \mathbf{x}_1, \dots, \mathbf{x}_n)]_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\text{AML}}^w} = \mathbf{0}^T \quad (3)$$

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

$$[\partial_{\boldsymbol{\theta}} J_{\text{AML}}(\boldsymbol{\theta}; \mathbf{x}_1, \dots, \mathbf{x}_n)]^T = 2\mathbf{X}_{\boldsymbol{\theta}}\boldsymbol{\theta}, \quad (4)$$

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

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

Thus (3) can be written as

$$[\mathbf{X}_{\boldsymbol{\theta}}\boldsymbol{\theta}]_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\text{AML}}^w} = \mathbf{0}. \quad (5)$$

An algorithm for numerically solving this equation proposed in [4] exploits the fact that a vector $\boldsymbol{\theta}$ satisfies (5) if and only if it is a solution of the *ordinary* eigenvalue problem

$$\mathbf{X}_{\boldsymbol{\theta}}\boldsymbol{\xi} = \lambda\boldsymbol{\xi} \quad (6)$$

corresponding to the eigenvalue $\lambda = 0$. Thus if $\boldsymbol{\theta}_{k-1}$ is an 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 [4] implementing this idea is presented in Figure 1. The scheme is seeded with the *algebraic least squares (ALS) estimate*, $\widehat{\boldsymbol{\theta}}_{\text{ALS}}$, defined as the unconstrained minimiser of the cost function $J_{\text{ALS}}(\boldsymbol{\theta}; \mathbf{x}_1, \dots, \mathbf{x}_n) = \|\boldsymbol{\theta}\|^{-2} \sum_{i=1}^n \boldsymbol{\theta}^T \mathbf{A}_i \boldsymbol{\theta}$, with $\|\boldsymbol{\theta}\| = (\sum_{j=1}^l \theta_j^2)^{1/2}$. The estimate $\widehat{\boldsymbol{\theta}}_{\text{ALS}}$ 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 $\boldsymbol{\theta}_0 = \widehat{\boldsymbol{\theta}}_{\text{ALS}}$.
- 2) Assuming that $\boldsymbol{\theta}_{k-1}$ is known, compute the matrices $\mathbf{M}_{\boldsymbol{\theta}_{k-1}}$ and $\mathbf{N}_{\boldsymbol{\theta}_{k-1}}$.
- 3) Compute a normalised eigenvector of the eigenvalue problem

$$\mathbf{M}_{\boldsymbol{\theta}_{k-1}}\boldsymbol{\xi} = \lambda\mathbf{N}_{\boldsymbol{\theta}_{k-1}}\boldsymbol{\xi}$$

corresponding to the eigenvalue closest to 1 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.

Fig. 2. Basic HEIV scheme.

III. BASIC HEIV SCHEME

A different method for numerically solving (5) was proposed by Leedan and Meer [10] and further extended by Matei and Meer [12], [11]. In one form¹ it relies upon re-expressing $\mathbf{X}_{\boldsymbol{\theta}}$ as

$$\mathbf{X}_{\boldsymbol{\theta}} = \mathbf{M}_{\boldsymbol{\theta}} - \mathbf{N}_{\boldsymbol{\theta}}$$

with

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

and restating the variational equation (5) as

$$\mathbf{M}_{\boldsymbol{\theta}}\boldsymbol{\theta} = \mathbf{N}_{\boldsymbol{\theta}}\boldsymbol{\theta}, \quad (7)$$

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

$$\mathbf{M}_{\boldsymbol{\theta}}\boldsymbol{\xi} = \lambda\mathbf{N}_{\boldsymbol{\theta}}\boldsymbol{\xi} \quad (8)$$

corresponding to the eigenvalue $\lambda = 1$. The heteroscedastic errors-in-variables scheme [10], [12], [11] (see Section VII for a clue as to the name) is an algorithm for solving (7) that exploits the above eigenvalue problem in a manner analogous to that in which FNS utilises the eigenvalue problem (6). The details are given in Figure 2.

As is easily seen from (12) below, the null space of each matrix \mathbf{B}_i contains the length l vector $[0, \dots, 0, 1]^T$. Consequently, $\mathbf{N}_{\boldsymbol{\theta}}$ is degenerate and solving (8) directly may be fraught with numerical instability problems. One way to get around this difficulty is to reduce the eigenvector problem (8) to a similar problem involving a positive-definitive right-hand side matrix. Such reduction is best achieved by reformulating the variational equation (7). This is described next.

¹This form of the method differs from the original form and is adopted to provide a convenient framework for subsequent considerations.

IV. REDUCED VARIATIONAL EQUATION

The vector $\mathbf{u}(\mathbf{x})$ has one entry equal to 1 and can be written as

$$\mathbf{u}(\mathbf{x}) = [\mathbf{z}(\mathbf{x})^T, 1]^T, \quad (9)$$

where $\mathbf{z}(\mathbf{x})$ is a ‘pure measurement’ vector of length $l-1$. The vector of parameters $\boldsymbol{\theta}$ can be partitioned conformally as

$$\boldsymbol{\theta} = [\boldsymbol{\eta}^T, \alpha]^T \quad (10)$$

with $\boldsymbol{\eta}$ a length $l-1$ vector and α a scalar. We are going to show that the variational equation (7) is equivalent to a system of two equations, one of which involves only $\boldsymbol{\eta}$ and can be solved in isolation, and the other expresses α in terms of $\boldsymbol{\eta}$. The first equation will lead to a desired non-degenerate eigenvalue problem.

We begin by noting that, in view of (9),

$$\mathbf{A}_i = \mathbf{u}(\mathbf{x}_i)\mathbf{u}(\mathbf{x}_i)^T = \begin{bmatrix} \mathbf{z}_i\mathbf{z}_i^T & \mathbf{z}_i \\ \mathbf{z}_i^T & 1 \end{bmatrix} \quad (11)$$

for each $i = 1, \dots, n$; here, of course, \mathbf{z}_i is short for $\mathbf{z}(\mathbf{x}_i)$. Another consequence of (9) is the identity

$$\partial_{\mathbf{x}}\mathbf{u}(\mathbf{x}) = \begin{bmatrix} \partial_{\mathbf{x}}\mathbf{z}(\mathbf{x}) \\ \mathbf{0}^T \end{bmatrix},$$

which implies that, for each $i = 1, \dots, n$,

$$\mathbf{B}_i = \begin{bmatrix} \mathbf{B}_i^0 & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix}, \quad (12)$$

with

$$\mathbf{B}_i^0 = \partial_{\mathbf{x}}\mathbf{z}(\mathbf{x}_i)\mathbf{A}_{\mathbf{x}_i}\partial_{\mathbf{x}}\mathbf{z}(\mathbf{x}_i)^T.$$

For each $i = 1, \dots, n$, define a weight

$$\beta_i = \frac{1}{\boldsymbol{\eta}^T \mathbf{B}_i^0 \boldsymbol{\eta}} \quad (13)$$

that depends on the data, their covariances, and the parameter vector $\boldsymbol{\eta}$. Let $\tilde{\mathbf{z}}$ be the centroid of the \mathbf{z}_i given by

$$\tilde{\mathbf{z}} = \frac{\sum_{i=1}^n \beta_i \mathbf{z}_i}{\sum_{i=1}^n \beta_i}, \quad (14)$$

and, for each $i = 1, \dots, n$, let

$$\mathbf{z}'_i = \mathbf{z}_i - \tilde{\mathbf{z}} \quad (15)$$

be the i th pure measurement vector relative to $\tilde{\mathbf{z}}$. Define two $(l-1) \times (l-1)$ matrices

$$\begin{aligned} \mathbf{M}'_{\boldsymbol{\eta}} &= \sum_{i=1}^n \beta_i \mathbf{z}'_i \mathbf{z}'_i{}^T, \\ \mathbf{N}'_{\boldsymbol{\eta}} &= \sum_{i=1}^n (\beta_i \mathbf{z}'_i{}^T \boldsymbol{\eta})^2 \mathbf{B}_i^0. \end{aligned}$$

If we also introduce

$$\mathbf{A}_i^0 = \mathbf{z}'_i \mathbf{z}'_i{}^T$$

we can then rewrite the matrices $\mathbf{M}'_{\boldsymbol{\eta}}$ and $\mathbf{N}'_{\boldsymbol{\eta}}$ as

$$\begin{aligned} \mathbf{M}'_{\boldsymbol{\eta}} &= \sum_{i=1}^n \frac{\mathbf{A}_i^0}{\boldsymbol{\eta}^T \mathbf{B}_i^0 \boldsymbol{\eta}}, \\ \mathbf{N}'_{\boldsymbol{\eta}} &= \sum_{i=1}^n \frac{\boldsymbol{\eta}^T \mathbf{A}_i^0 \boldsymbol{\eta}}{(\boldsymbol{\eta}^T \mathbf{B}_i^0 \boldsymbol{\eta})^2} \mathbf{B}_i^0, \end{aligned}$$

revealing their resemblance to $\mathbf{M}_{\boldsymbol{\theta}}$ and $\mathbf{N}_{\boldsymbol{\theta}}$. The choice of the weights β_i is largely motivated by the desire to achieve such a resemblance in the first place. Obviously, $\mathbf{M}'_{\boldsymbol{\eta}}$ and $\mathbf{N}'_{\boldsymbol{\eta}}$ both depend not only on $\boldsymbol{\eta}$ but also on the data and their covariances.

We now show that $\boldsymbol{\theta} = [\boldsymbol{\eta}^T, \alpha]^T$ satisfies (7) if and only if the following system of equations holds:

$$\mathbf{M}'_{\boldsymbol{\eta}} \boldsymbol{\eta} = \mathbf{N}'_{\boldsymbol{\eta}} \boldsymbol{\eta}, \quad (16)$$

$$\alpha = -\tilde{\mathbf{z}}^T \boldsymbol{\eta}. \quad (17)$$

Note that the first equation constrains solely $\boldsymbol{\eta}$ and therefore can be solved separately. Once $\boldsymbol{\eta}$ is determined, α is readily prescribed by the second equation. As will become apparent shortly, the equations decouple as a result of the specific choice of the centroid $\tilde{\mathbf{z}}$. Of the two constraints, the first plays a leading role and will be called the *reduced variational equation*. A key feature of this equation is that its right-hand side matrix $\mathbf{N}'_{\boldsymbol{\eta}}$, unlike $\mathbf{N}_{\boldsymbol{\theta}}$, is generically *positive definite* if $n \geq l$.

To show the equivalence of (7) and the system comprising (16) and (17), first note that, by (10) and (12), for each $i = 1, \dots, n$,

$$\boldsymbol{\theta}^T \mathbf{B}_i \boldsymbol{\theta} = \boldsymbol{\eta}^T \mathbf{B}_i^0 \boldsymbol{\eta}$$

and further, by (13),

$$\beta_i = \frac{1}{\boldsymbol{\theta}^T \mathbf{B}_i \boldsymbol{\theta}}. \quad (18)$$

Consequently,

$$\mathbf{M}_{\boldsymbol{\theta}} = \sum_{i=1}^n \beta_i \mathbf{A}_i. \quad (19)$$

Observe next that, in view of (12),

$$\mathbf{N}_{\boldsymbol{\theta}} = \begin{bmatrix} \mathbf{N}_{\boldsymbol{\theta}}^0 & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} \quad (20)$$

with

$$\mathbf{N}_{\boldsymbol{\theta}}^0 = \sum_{i=1}^n \frac{\boldsymbol{\theta}^T \mathbf{A}_i \boldsymbol{\theta}}{(\boldsymbol{\theta}^T \mathbf{B}_i \boldsymbol{\theta})^2} \mathbf{B}_i^0.$$

Now if $\boldsymbol{\theta} = [\boldsymbol{\eta}^T, \alpha]^T$ satisfies (7), then, in view of (11), (19) and (20), (7) can be rewritten as

$$\sum_{i=1}^n \beta_i \begin{bmatrix} \mathbf{z}_i \mathbf{z}_i^T & \mathbf{z}_i \\ \mathbf{z}_i^T & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\eta} \\ \alpha \end{bmatrix} = \begin{bmatrix} \mathbf{N}_{\boldsymbol{\theta}}^0 & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\eta} \\ \alpha \end{bmatrix},$$

or equivalently as the system

$$\sum_{i=1}^n \beta_i (\alpha \mathbf{z}_i + \mathbf{z}_i \mathbf{z}_i^T \boldsymbol{\eta}) = \mathbf{N}_{\boldsymbol{\theta}}^0 \boldsymbol{\eta}, \quad (21)$$

$$\sum_{i=1}^n \beta_i (\alpha + \mathbf{z}_i^T \boldsymbol{\eta}) = 0. \quad (22)$$

On account of (14) and (22),

$$\alpha + \tilde{\mathbf{z}}^T \boldsymbol{\eta} = \frac{\sum_{i=1}^n \beta_i (\alpha + \mathbf{z}_i^T \boldsymbol{\eta})}{\sum_{i=1}^n \beta_i} = 0, \quad (23)$$

and this immediately yields (17). To show that (16) also holds, note that, by (9) and (10), for each $i = 1, \dots, n$,

$$\mathbf{u}(\mathbf{x}_i)^T \boldsymbol{\theta} = \alpha + \mathbf{z}_i^T \boldsymbol{\eta},$$

and, by (15) and (17),

$$\alpha + \mathbf{z}_i^T \boldsymbol{\eta} = (\mathbf{z}_i - \tilde{\mathbf{z}})^T \boldsymbol{\eta} = \mathbf{z}'_i{}^T \boldsymbol{\eta}.$$

Hence

$$\boldsymbol{\theta}^T \mathbf{A}_i \boldsymbol{\theta} = (\mathbf{u}(\mathbf{x}_i)^T \boldsymbol{\theta})^2 = (\mathbf{z}'_i{}^T \boldsymbol{\eta})^2.$$

This together with (18) implies that

$$\frac{\boldsymbol{\theta}^T \mathbf{A}_i \boldsymbol{\theta}}{(\boldsymbol{\theta}^T \mathbf{B}_i \boldsymbol{\theta})^2} = (\beta_i \mathbf{z}'_i{}^T \boldsymbol{\eta})^2,$$

whence immediately

$$\mathbf{N}_{\boldsymbol{\theta}}^0 = \mathbf{N}'_{\boldsymbol{\eta}}.$$

With this identity, (21) can be rewritten as

$$\sum_{i=1}^n \beta_i (\alpha \mathbf{z}_i + \mathbf{z}_i \mathbf{z}_i^T \boldsymbol{\eta}) = \mathbf{N}'_{\boldsymbol{\eta}} \boldsymbol{\eta}. \quad (24)$$

On the other hand, taking into account (15), we see that

$$\begin{aligned} \sum_{i=1}^n \beta_i (\alpha \mathbf{z}_i + \mathbf{z}_i \mathbf{z}_i^T \boldsymbol{\eta}) &= \sum_{i=1}^n \beta_i \mathbf{z}_i (\alpha + \mathbf{z}_i^T \boldsymbol{\eta}) \\ &= \sum_{i=1}^n \beta_i (\mathbf{z}'_i + \tilde{\mathbf{z}}) (\alpha + \mathbf{z}_i^T \boldsymbol{\eta}). \end{aligned} \quad (25)$$

By (22),

$$\sum_{i=1}^n \beta_i \tilde{\mathbf{z}} (\alpha + \mathbf{z}_i^T \boldsymbol{\eta}) = \tilde{\mathbf{z}} \sum_{i=1}^n \beta_i (\alpha + \mathbf{z}_i^T \boldsymbol{\eta}) = \mathbf{0}, \quad (26)$$

and by (15) and (23),

$$\begin{aligned} \sum_{i=1}^n \beta_i \mathbf{z}'_i (\alpha + \mathbf{z}_i^T \boldsymbol{\eta}) &= \sum_{i=1}^n \beta_i \mathbf{z}'_i (\alpha + \tilde{\mathbf{z}}^T \boldsymbol{\eta} + \mathbf{z}'_i{}^T \boldsymbol{\eta}) \\ &= \sum_{i=1}^n \beta_i \mathbf{z}'_i \mathbf{z}'_i{}^T \boldsymbol{\eta} = \mathbf{M}'_{\boldsymbol{\eta}} \boldsymbol{\eta}. \end{aligned} \quad (27)$$

Combining (25), (26), and (27), we obtain

$$\sum_{i=1}^n \beta_i (\alpha \mathbf{z}_i + \mathbf{z}_i \mathbf{z}_i^T \boldsymbol{\eta}) = \mathbf{M}'_{\boldsymbol{\eta}} \boldsymbol{\eta}.$$

This jointly with (24) finally yields (16), as desired.

Working backwards, one can easily infer that if $\boldsymbol{\eta}$ satisfies (16) and α is given by (17), then $\boldsymbol{\theta} = [\boldsymbol{\eta}^T, \alpha]^T$ satisfies (7).

- 1) Set $\boldsymbol{\eta}_0 = \hat{\boldsymbol{\eta}}_{\text{ALS}}$.
- 2) Assuming that $\boldsymbol{\eta}_{k-1}$ is known, compute the matrices $\mathbf{M}'_{\boldsymbol{\eta}_{k-1}}$ and $\mathbf{N}'_{\boldsymbol{\eta}_{k-1}}$.
- 3) Compute a normalised eigenvector of the eigenvalue problem

$$\mathbf{M}'_{\boldsymbol{\eta}_{k-1}} \boldsymbol{\zeta} = \lambda \mathbf{N}'_{\boldsymbol{\eta}_{k-1}} \boldsymbol{\zeta}$$

corresponding to the eigenvalue closest to 1 and take this eigenvector for $\boldsymbol{\eta}_k$.

- 4) If $\boldsymbol{\eta}_k$ is sufficiently close to $\boldsymbol{\eta}_{k-1}$, then terminate the procedure; otherwise increment k and return to Step 2.

Fig. 3. Reduced HEIV scheme.

V. REDUCED HEIV SCHEME

The algebraic least squares estimates $\hat{\boldsymbol{\eta}}_{\text{ALS}}$ and $\hat{\alpha}_{\text{ALS}}$ are naturally defined as the respective components in the representation

$$\hat{\boldsymbol{\theta}}_{\text{ALS}} = [(\hat{\boldsymbol{\eta}}_{\text{ALS}})^T, \hat{\alpha}_{\text{ALS}}]^T.$$

Analogously the weak approximate maximum likelihood estimates $\hat{\boldsymbol{\eta}}_{\text{AML}}^w$ and $\hat{\alpha}_{\text{AML}}^w$ are defined via the decomposition

$$\hat{\boldsymbol{\theta}}_{\text{AML}}^w = [(\hat{\boldsymbol{\eta}}_{\text{AML}}^w)^T, \hat{\alpha}_{\text{AML}}^w]^T.$$

In view of (17), $\hat{\alpha}_{\text{AML}}^w$ is uniquely determined by $\hat{\boldsymbol{\eta}}_{\text{AML}}^w$: when the centroid $\tilde{\mathbf{z}}$ is taken with the weights

$$\beta_i = \frac{1}{(\hat{\boldsymbol{\eta}}_{\text{AML}}^w)^T \mathbf{B}_i^0 \hat{\boldsymbol{\eta}}_{\text{AML}}^w},$$

$\hat{\alpha}_{\text{AML}}^w$ is given by

$$\hat{\alpha}_{\text{AML}}^w = -\tilde{\mathbf{z}}^T \hat{\boldsymbol{\eta}}_{\text{AML}}^w.$$

Now, the generalised eigenvalue problem

$$\mathbf{M}'_{\boldsymbol{\eta}} \boldsymbol{\zeta} = \lambda \mathbf{N}'_{\boldsymbol{\eta}} \boldsymbol{\zeta}, \quad (28)$$

is non-degenerate: the matrix $\mathbf{N}'_{\boldsymbol{\eta}}$ is positive-definite. Accordingly, $\hat{\boldsymbol{\eta}}_{\text{AML}}^w$ can be determined with use of a simple modification of the HEIV algorithm. The steps of this *reduced* HEIV scheme are given in Figure 3. It is essentially in this form that the HEIV algorithm was originally advanced by its inventors [10], [12], [11], who also proposed a robust procedure for solving the eigenvalue problem (28) based upon *generalised singular value decomposition* (GSVD) of a pair of matrices.

It is worthy of noting that reduction to a non-degenerate form of eigenvalue problems similar to (28) is crucial for computing some other types of estimates, notably Kanatani-like renormalisation estimates [3] [9, Chap. 9] (these are approximates of $\hat{\boldsymbol{\theta}}_{\text{AML}}^w$ of some sort), and ellipse-specific estimates obtainable with use of the improved version of the direct least-squares fitting algorithm of Fitzgibbon et al. [7], due to Halíř and Flusser [8].

VI. ALTERNATIVE INITIALISATION

The HEIV scheme, as described by Leedan and Meer [10], is seeded by an initial estimate which is the solution of the generalised eigenvalue problem

$$A^0 \zeta = \lambda B^0 \zeta$$

corresponding to the smallest eigenvalue, with B^0 defined as a component of the minimiser of the function

$$(\gamma_1, \dots, \gamma_n, B^0) \mapsto \sum_{i=1}^n \|B_i^0 - \gamma_i B^0\|_F^2,$$

where $\gamma_1, \dots, \gamma_n$ are scalars and $\|\cdot\|_F$ is the Frobenius norm. The minimiser $(\gamma_1, \dots, \gamma_n, B^0)$ is characterised by the (variational) equations

$$B^0 = \frac{\sum_{i=1}^n \gamma_i B_i^0}{\sum_{i=1}^n \gamma_i^2}, \quad (29)$$

$$\gamma_i = \frac{\text{tr}(B_i^0 B^0)}{\|B^0\|_F^2} \quad (1 \leq i \leq n), \quad (30)$$

where tr denotes trace. The above system cannot be solved in closed form, but an approximate solution can be obtained in steps as follows:

- assume $\gamma_i = 1$ for each $i = 1, \dots, n$,
- compute B^0 using (29),
- recompute the γ_i employing (30),
- recompute B^0 using (29).

According to Leedan and Meer, the accuracy of the initial seed is not crucial, and the HEIV method converges successfully, even when seeded with a random initial estimate.

VII. ORIGINS OF HEIV

The original derivation of the HEIV algorithm, as given in [10], [12], [11], is different from the one presented here. In our exposition, the core of HEIV, namely its reduced form, results from reformulating the variational equation so that the associated generalised eigenvalue problem becomes non-degenerate. The original derivation is based on a direct application of the maximum likelihood principle to a statistical model operating with candidate probability distributions for $z(\mathbf{x}_1), \dots, z(\mathbf{x}_n)$, with the random variables $\mathbf{x}_1, \dots, \mathbf{x}_n$ modelling the image data $\mathbf{x}_1, \dots, \mathbf{x}_n$. When the \mathbf{x}_i have equal variances (as is often assumed), the $z(\mathbf{x}_i)$ form, as a rule, a *heteroscedastic* set of random variables, that is having different variances. This explains the term ‘heteroscedastic’ in the name of the HEIV algorithm. The ‘errors-in-variables’ part of the label alludes to the adopted statistical model being a so-called errors-in-variables model—the scalar components of each \mathbf{x}_i are not segregated into two exclusive groups of explanatory (essentially non-random) and response (random) variables, and are all consistently treated as random variables.

VIII. EXPERIMENTS

Relative performance of the FNS and HEIV methods was experimentally assessed by running a series of simulations involving synthetic data. The particular problem considered was

estimation of epipolar geometry. Here, data was presented in the form of matched corresponding points from left and right images of a stereo pair, and the goal was to estimate the associated *fundamental matrix*. Details of the various expressions involved are presented elsewhere [4].

In these experiments, we tested five estimation methods, which we denote as ALS, FNS, HB, HR, and HEIV. ALS is the simple, direct algebraic least squares method described in Section II. It is included as a method of a different category to give a sense of scale to the forthcoming numerical results. The FNS, HB, and HR methods were implemented as specified in Figures 1, 2 and 3, respectively. These iterative methods were terminated when the difference in norm between successive estimates was less than a common, very small threshold. Estimates of the final method, HEIV, were obtained using the MATLAB source code supplied by the authors of the original HEIV papers.²

The simulations were based on a set of ‘true’ pairs of corresponding points generated by selecting a realistic stereo camera configuration, randomly choosing many 3D points, and projecting the 3D points onto two image planes. Only those scene points were considered that had both projections confined to the image size of 1000×1000 pixels.

For each of $N = 5000$ iterations, the true corresponding points were perturbed by homogeneous Gaussian noise to produce noisy points. These noisy points were then used to generate a fundamental matrix estimate for each of the five estimation methods. For each estimate, the value of the J_{AML} cost function was computed. Comparison was undertaken in this realm as J_{AML} is the basis for our rationalising and linking of the various iterative methods considered. Note that the rank-2 constraint was not imposed as this would otherwise obfuscate comparison (the constraint is usually implemented as a separate post-process). In these tests, the level of noise was fixed at $\sigma = 1.0$ pixels, although similar results were obtained using different noise levels.

Figure 4 shows the histograms of J_{AML} values associated with each of the estimators (the average value is given in the top right hand corner of each histogram). In contrast with the ALS method, the iterative methods generate very similar response profiles. Table I compares estimators pair-wise by showing both the maximum and average differences in associated J_{AML} values over the complete set of trials. The respective top left elements are computed via the expressions $\max_{i=1..N} |J_{\text{AML}}(\theta_{\text{HEIV}}^i) - J_{\text{AML}}(\theta_{\text{ALS}}^i)|$ and $\sum_{i=1}^N |J_{\text{AML}}(\theta_{\text{HEIV}}^i) - J_{\text{AML}}(\theta_{\text{ALS}}^i)|$. The results demonstrate that the methods FNS, HB, HR, and HEIV deliver estimates whose associated J_{AML} values are extremely close. As would be expected from the earlier theory, the HR and HEIV methods prove to be almost numerically identical.

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.

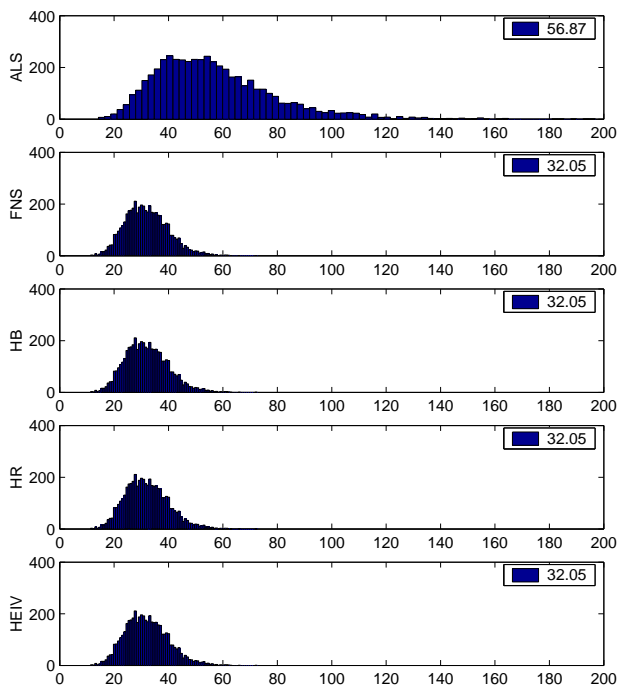
²<http://www.caip.rutgers.edu/riul/research/code.html>

<i>Max. diff.</i>	ALS	FNS	HB	HR
HEIV	146	4.7×10^{-6}	7.5×10^{-5}	2.6×10^{-8}
HR	146	4.7×10^{-6}	7.5×10^{-5}	
HB	146	7.1×10^{-5}		
FNS	146			

<i>Avg. diff.</i>	ALS	FNS	HB	HR
HEIV	24.8	5.7×10^{-8}	2.0×10^{-6}	2.5×10^{-10}
HR	24.8	5.8×10^{-8}	2.0×10^{-6}	
HB	24.8	2.0×10^{-6}		
FNS	24.8			

TABLE I

MAXIMUM AND AVERAGE DIFFERENCES IN J_{AML} VALUES FOR DIFFERENT ESTIMATION METHODS.

Fig. 4. Histograms of J_{AML} values for various methods

- [2] M. J. Brooks, W. Chojnacki, D. Gawley, and A. van den Hengel. What value covariance information in estimating vision parameters? In *Proceedings, Eighth IEEE International Conference of Computer Vision, Vancouver, British Columbia, Canada, July 7–14, 2001*, volume 1, pages 302–308, Los Alamitos, California, 2001. IEEE Computer Society.
- [3] W. Chojnacki, M. J. Brooks, and A. van den Hengel. Rationalising the renormalisation method of Kanatani. *Journal of Mathematical Imaging and Vision*, 14(1):21–38, 2001.
- [4] W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley. On the fitting of surfaces to data with covariances. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1294–1303, 2000.
- [5] W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley. A new constrained parameter estimator: experiments in fundamental matrix computation. In *Proceedings of the 2002 British Machine Vision Conference, Cardiff, UK, Sept. 2002*.
- [6] O. D. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge, Mass., 1993.
- [7] A. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct least square fitting of ellipses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*,

21(5):476–480, 1999.

- [8] R. Halff and J. Flusser. Numerically stable direct least squares fitting of ellipses. In V. Skala, editor, *The Sixth International Conference in Central Europe on Computer Graphics and Visualization, WSCG '98, Conference Proceedings, University of West Bohemia, Plzeň, Czech Republic, February 9–13, 1998*, volume 1, pages 125–132, 1998. Available at <http://wscg.zcu.cz/wscg98/wscg98.htm>.
- [9] K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice*. Elsevier, Amsterdam, 1996.
- [10] Y. Leedan and P. Meer. Heteroscedastic regression in computer vision: problems with bilinear constraint. *International Journal of Computer Vision*, 37(2):127–150, 2000.
- [11] 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. Available at <http://www.caip.rutgers.edu/riul/research/theses.html>.
- [12] B. Matei and P. Meer. A general method for errors-in-variables problems in computer vision. In *Proceedings, CVPR 2000, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, June 13–15, 2000*, volume 2, pages 18–25, Los Alamitos, CA, 2000. IEEE Computer Society Press.