

# CONSTRAINED GENERALISED PRINCIPAL COMPONENT ANALYSIS

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

School of Computer Science, University of Adelaide, Adelaide, SA 5005, Australia  
CRC for Sensor Signal and Information Processing, Mawson Lakes, SA 5095, Australia

{wojtek, anton, mjb}@cs.adelaide.edu.au

## ABSTRACT

Generalised Principal Component Analysis (GPCA) is a recently devised technique for fitting a multi-component structure to data. Unlike other methods which intertwine the processes of estimating structure components and segmenting data points into clusters associated with putative components, GPCA estimates a multi-component structure with no recourse to data clustering. The standard GPCA algorithm searches for an estimate by minimising an appropriate misfit function. The underlying constraints on the model parameters are ignored. Here we promote a variant of GPCA that incorporates the parameter constraints and exploits constrained rather than unconstrained minimisation of the error function. The output of any GPCA algorithm hardly ever perfectly satisfies the parameter constraints. The new version of GPCA greatly facilitates the final correction of the algorithm output to satisfy perfectly the constraints, making this step less prone to error in the presence of noise. The method is applied to the problem of fitting a pair of lines to noisy image points, but has potential for use in more general multi-component structure fitting.

## 1. INTRODUCTION

One of the challenges of image analysis and computer vision is to develop effective ways to fit a multi-component structure to data. A classical example problem is fitting multiple lines to data [1, 2]. Several methods have been proposed for solving this particular task, including those based on the Hough transform [3], K-means, EM, LMedS and RANSAC algorithms [4]. More recently, there has been interest in fitting multiple linear manifolds to data. This more general problem arose in the analysis of dynamical scenes in computer vision in connection with the recovery of multiple motion models from image data [5, 6]. To tackle it, a new approach has been put forth under the label of the *generalised principle component analysis* (GPCA) [7, 8]. The GPCA method employs a parametric model in which parameters describe a multi-component structure to which

various parts of a data set adhere. The number of components is assumed to be fixed and known beforehand. The relationship between data and components is encoded in a multivariate polynomial. In the special, but representative, case of multi-line fitting, the components are lines, the order of the polynomial coincides with the number of the components, and the recovery of the components is achieved by factoring the polynomial into a product of multivariate monomials, each corresponding to a separate line. The success of the whole procedure rests upon generation of a meaningful polynomial to factor.

This paper presents a variant of GPCA which advocates the use of *constrained* optimisation as a crucial step in component recovery. We concentrate on a particular problem of fitting two lines to data as in this case the underlying analysis is particularly simple and illuminating. Notwithstanding the specificity of our presentation, the multi-line and, more generally, multi-component fitting problems can be treated—upon suitable modification—within the same general framework.

At the technical level, the contribution of the paper is three-fold. First, it gives a statistically sound *cost function* measuring how well a given model instance describes the data. The cost function is evolved by applying the maximum likelihood principle to a Gaussian model of errors in the data. Second, a pair of lines is shown to be effectively estimated by minimising the cost function subject to a certain *parameter constraint*. A novel iterative method for computing a constrained minimiser is given. Finally, a simple method is presented for converting nearly optimal estimates obtained by iterative constrained optimisation techniques (hyperbolae with high eccentricity) into estimates representing a *correct geometric structure* (pairs of lines).

The original GPCA algorithm [6, 7] employs algebraic factorisation of a multivariate polynomial whose coefficients are obtained via *unconstrained* minimisation of a cost function equivalent to the one used in the present paper. The method does not require data segmentation and in that differs from iterative methods like K-means and EM which alternate between estimating structure components and grouping the data around individual components. How-

---

This work was supported by the Australian Research Council.

ever, because of its reliance on computation of roots of polynomials—a numerically fragile operation—the GPCA algorithm is sensitive to noise. To curb adverse effects of noise, the subsequent version of GPCA [5, 8] uses polynomial differentiation instead of polynomial factorisation, but at the cost of employing some form of data segmentation—one data point per component is needed to effectuate the estimation step.

The present paper shows—and this is its main conceptual contribution—that the approach taken by the original version of GPCA can be sustained even in the presence of moderate noise if unconstrained minimisation is replaced by constrained minimisation. We demonstrate empirically that constrained optimisation leads, in practice, to estimates that can be encoded into nearly factorisable polynomials. These estimates can be upgraded to estimates corresponding to perfectly factorisable polynomials by means of a simple correction procedure. Because a minor adjustment of the unconstrained minimiser is needed, the upgrading procedure operates reliably. Rather than use polynomial factorisation, the correction procedure in our version of the GPCA involves singular value decomposition. Its simple form reflects the special nature of the estimation problem considered.

The estimate obtained by applying the method presented in the paper represents a pair of lines and as such is an instance of a conic—a degenerate conic. Thus, effectively, our variant of GPCA is a method for degenerate-conic fitting and can be viewed as an addition to the growing body of algorithms for fitting to data a conic of a type specified in advance [9–12].

## 2. BACKGROUND

A *line* is a locus of points  $\mathbf{x} = [m_1, m_2]^T$  in the Euclidean plane  $\mathbb{R}^2$  satisfying the equation

$$l_1 m_1 + l_2 m_2 + l_3 = 0.$$

Employing homogeneous coordinates  $\mathbf{m} = [m_1, m_2, 1]^T$  and  $\mathbf{l} = [l_1, l_2, l_3]^T$ , the same line can be identified with the subset of the projective plane  $\mathbb{P}^2$  given by  $Z_l = \{\mathbf{m} \in \mathbb{P}^2 \mid \mathbf{l}^T \mathbf{m} = 0\}$ . A *conic* is a locus of points  $\mathbf{x} = [m_1, m_2]^T$  satisfying the equation

$$am_1^2 + bm_1 m_2 + cm_2^2 + dm_1 + em_2 + f = 0,$$

where  $a, b$  and  $c$  are not all zero. Introducing the symmetric matrix  $\mathbf{C}$

$$\mathbf{C} = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix},$$

the same conic can be described as  $Z_C = \{\mathbf{m} \in \mathbb{P}^2 \mid \mathbf{m}^T \mathbf{C} \mathbf{m} = 0\}$ . A *non-degenerate* conic satisfies  $\det \mathbf{C} \neq$

0 and is either an ellipse, or a parabola, or a hyperbola depending on whether the *discriminant*  $\Delta = b^2 - 4ac$  is negative, zero or positive. If  $\det \mathbf{C} = 0$ , then the conic is *degenerate*. A degenerate conic represents either two intersecting lines, a (double) line, or a point, as we now critically recall.

A union of two lines,  $Z_{l_1} \cup Z_{l_2}$ , obeys

$$\mathbf{l}_1^T \mathbf{m} \cdot \mathbf{l}_2^T \mathbf{m} = \mathbf{m}^T \mathbf{l}_1 \mathbf{l}_2^T \mathbf{m} = 0$$

or equivalently, given that  $\mathbf{m}^T \mathbf{l}_1 \mathbf{l}_2^T = \mathbf{m}^T \mathbf{l}_2 \mathbf{l}_1^T \mathbf{m} = 0$ ,

$$\mathbf{m}^T (\mathbf{l}_1 \mathbf{l}_2^T + \mathbf{l}_2 \mathbf{l}_1^T) \mathbf{m} = 0. \quad (1)$$

With  $\mathbf{C} = \mathbf{l}_1 \mathbf{l}_2^T + \mathbf{l}_2 \mathbf{l}_1^T$ , a symmetric matrix, the above equation can be rewritten as  $\mathbf{m}^T \mathbf{C} \mathbf{m} = 0$ , showing that  $Z_{l_1} \cup Z_{l_2}$  is identical with the conic  $Z_C$ . The matrices  $\mathbf{l}_i \mathbf{l}_j^T$  are rank-1, so the rank of  $\mathbf{C}$  is no greater than 2 and the conic is degenerate. If  $\mathbf{l}_1 = \mathbf{l}_2$ , then  $Z_C$  represents a single, repeated line; in this case the conic equation  $(\mathbf{l}_1^T \mathbf{m})^2 = 0$  is equivalent to the line equation  $\mathbf{l}_1^T \mathbf{m} = 0$ . Finally, a point  $[p_1, p_2]^T$  can be represented as the degenerate conic  $(m_1 - p_1)^2 + (m_2 - p_2)^2 = 0$  corresponding to

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & -p_1 \\ 0 & 1 & -p_2 \\ -p_1 & -p_2 & p_1^2 + p_2^2 \end{bmatrix}.$$

To see that a pair of lines, a double line and a point are the only possible types of degenerate conic, suppose that  $\mathbf{C}$  is a non-zero symmetric singular matrix. Then  $\mathbf{C}$  admits an *eigenvalue decomposition* (EVD) of the form  $\mathbf{C} = \mathbf{V} \mathbf{D} \mathbf{V}^T$ , where  $\mathbf{V}$  is an orthogonal  $3 \times 3$  matrix and  $\mathbf{D} = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$ , with  $\lambda_i$  ( $i = 1, 2, 3$ ) a real number [13]. The eigenvalue decomposition differs from the *singular value decomposition* (SVD) of  $\mathbf{C}$  in that the latter uses two orthogonal, possibly different, matrices  $\mathbf{U}$  and  $\mathbf{V}$ , and that the former uses a diagonal matrix whose entries are not necessarily non-negative. However, the EVD and SVD of the symmetric  $\mathbf{C}$  are closely related—any of the two orthogonal factors  $\mathbf{U}, \mathbf{V}$  in the SVD can serve as  $\mathbf{V}$  in the EVD, and  $\mathbf{D}$  in the EVD can be obtained from the diagonal factor in the SVD by placing a minus sign before each diagonal entry for which the corresponding columns in  $\mathbf{U}$  and  $\mathbf{V}$  differ by a sign, with all remaining entries left intact. For each  $i = 1, 2, 3$ , let  $\mathbf{v}_i$  be the  $i$ th column vector of  $\mathbf{V}$ . Then, clearly,  $\mathbf{v}_i$  is an eigenvector of  $\mathbf{C}$  corresponding to the eigenvalue  $\lambda_i$ ,  $\mathbf{C} \mathbf{v}_i = \lambda_i \mathbf{v}_i$ , and, moreover,  $\mathbf{C} = \sum_{i=1}^3 \lambda_i \mathbf{v}_i \mathbf{v}_i^T$ . Now  $\det \mathbf{C} = \lambda_1 \lambda_2 \lambda_3 = 0$  so one eigenvalue, say  $\lambda_3$ , is zero, implying that  $\mathbf{C} = \sum_{i=1}^2 \lambda_i \mathbf{v}_i \mathbf{v}_i^T$ . If another eigenvalue, say  $\lambda_2$ , is zero too, then  $\mathbf{C} = \lambda_1 \mathbf{v}_1 \mathbf{v}_1^T$  and, since the remaining eigenvalue,  $\lambda_1$ , has to be non-zero,  $Z_C$  coincides with the line  $Z_{\mathbf{v}_1}$ . If  $\lambda_3$  is the only zero eigenvalue, then there are two possibilities—either  $\lambda_1$  and  $\lambda_2$  are of same sign, or  $\lambda_1$  and  $\lambda_2$  are of opposite signs. In the first case  $Z_C$  reduces to the linear span of  $\mathbf{v}_3 = [v_{13}, v_{23}, v_{33}]^T$

and represents a single point in  $\mathbb{P}^2$ ; if  $v_{33} \neq 0$ , then this point is part of  $\mathbb{R}^2$  and is given by  $[v_{13}/v_{33}, v_{23}/v_{33}, 1]^T$ . In the other case,  $Z_C$  represents a pair of lines in  $\mathbb{P}^2$ . Indeed, without loss of generality, we may suppose that  $\lambda_1 > 0$  and  $\lambda_2 < 0$ . Then

$$\lambda_1 \mathbf{v}_1 \mathbf{v}_1^T + \lambda_2 \mathbf{v}_2 \mathbf{v}_2^T = \mathbf{l}_1 \mathbf{l}_1^T + \mathbf{l}_2 \mathbf{l}_2^T,$$

where  $\mathbf{l}_1 = \sqrt{\lambda_1} \mathbf{v}_1 + \sqrt{-\lambda_2} \mathbf{v}_2$  and  $\mathbf{l}_2 = \sqrt{\lambda_1} \mathbf{v}_1 - \sqrt{-\lambda_2} \mathbf{v}_2$ . Consequently,

$$\mathbf{m}^T \mathbf{C} \mathbf{m} = \mathbf{m}^T \mathbf{l}_1 \mathbf{l}_1^T \mathbf{m} + \mathbf{m}^T \mathbf{l}_2 \mathbf{l}_2^T \mathbf{m} = 2(\mathbf{l}_1^T \mathbf{m})(\mathbf{l}_2^T \mathbf{m}),$$

so  $Z_C$  is the union of the lines  $Z_{\mathbf{l}_1}$  and  $Z_{\mathbf{l}_2}$ . The identification of  $Z_C$  with  $Z_{\mathbf{l}_1} \cup Z_{\mathbf{l}_2}$  via the factorisation of the binomial  $\mathbf{m}^T \mathbf{C} \mathbf{m}$  as above exemplifies the general factorisation principle underlying GPCA.

### 3. ESTIMATION PROBLEM

The equation for a conic  $Z_C$  can alternatively be written as

$$\boldsymbol{\theta}^T \mathbf{u}(\mathbf{x}) = 0, \quad (2)$$

where  $\boldsymbol{\theta} = [\theta_1, \dots, \theta_6]^T = [a, b, c, d, e, f]^T$  and  $\mathbf{u}(\mathbf{x}) = [m_1^2, m_1 m_2, m_2^2, m_1, m_2, 1]^T$ . The singularity constraint  $\det \mathbf{C} = 0$  can be written as

$$\phi(\boldsymbol{\theta}) = 0, \quad (3)$$

where  $\phi(\boldsymbol{\theta}) = \theta_1 \theta_3 \theta_6 - \theta_1 \theta_2^2 / 4 - \theta_2^2 \theta_6 / 4 + \theta_2 \theta_4 \theta_5 / 4 - \theta_4^2 \theta_3 / 4$ . Note that  $\phi$  is *homogeneous* of degree 3—that is such that

$$\phi(t\boldsymbol{\theta}) = t^\kappa \phi(\boldsymbol{\theta}) \quad (4)$$

for every non-zero scalar  $t$ , with  $\kappa = 3$  the index of homogeneity.

Together, equations (2) and (3) form a parametric model that encapsulates the configuration comprising a pair of lines and a point at one of these lines. In this setting,  $\boldsymbol{\theta}$  is the vector of parameters representing the lines and  $\mathbf{x}$  is the *ideal* datum representing the point.

Associated with this model is the following estimation problem: Given a collection  $\mathbf{x}_1, \dots, \mathbf{x}_n$  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 (2) associated with  $\mathbf{x} = \mathbf{x}_i$  ( $i = 1, \dots, n$ ), find  $\boldsymbol{\theta} \neq \mathbf{0}$  satisfying (3) for which the cost function attains its minimum.

Use of the *Gaussian model of errors* in data in conjunction with the *principle of maximum likelihood* leads to the *approximated maximum likelihood (AML) 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 2 vector  $\mathbf{y}$ ,  $\partial_{\mathbf{x}} \mathbf{u}(\mathbf{y})$  denotes the  $6 \times 2$  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  $2 \times 2$  symmetric *covariance matrix* describing the uncertainty of the data point  $\mathbf{x}_i$  [14–16]. If  $J_{\text{AML}}$  is minimised over those non-zero parameter vectors for which (3) holds, then the vector at which the minimum of  $J_{\text{AML}}$  is attained, the constrained minimiser of  $J_{\text{AML}}$ , defines the *approximated maximum likelihood estimate*  $\hat{\boldsymbol{\theta}}_{\text{AML}}$ . The unconstrained minimiser of  $J_{\text{AML}}$  obtained by ignoring the constraint (3) and searching over all of the parameter space defines the *unconstrained approximated likelihood estimate*,  $\hat{\boldsymbol{\theta}}_{\text{AML}}^u$ . 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  $\phi$  is invariant to multiplication by non-zero scalars, so both  $\hat{\boldsymbol{\theta}}_{\text{AML}}$  and  $\hat{\boldsymbol{\theta}}_{\text{AML}}^u$  are determined only up to scale. Obviously,  $\hat{\boldsymbol{\theta}}_{\text{AML}}$  is the preferred estimate of  $\boldsymbol{\theta}$ , with  $\hat{\boldsymbol{\theta}}_{\text{AML}}^u$  being the second best choice.

### 4. UNCONSTRAINED MINIMISATION

The unconstrained minimiser  $\hat{\boldsymbol{\theta}}_{\text{AML}}^u$  satisfies the *optimality condition* for unconstrained minimisation

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

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

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

where

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

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

The optimality condition rewritten as

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

serves as the basis for isolating  $\hat{\boldsymbol{\theta}}_{\text{AML}}^u$ . Two Newton-like iterative algorithms can be used for solving (5). The *fundamental numerical scheme* (FNS) [15] exploits the fact that a vector  $\boldsymbol{\theta}$  satisfies (5) if and only if it is a solution of the *ordinary* eigenvalue problem

$$\mathbf{X}\boldsymbol{\theta}\boldsymbol{\xi} = \lambda \boldsymbol{\xi}$$

corresponding to the eigenvalue  $\lambda = 0$ . Given a current approximate solution  $\boldsymbol{\theta}_c$ , an updated solution  $\boldsymbol{\theta}_+$  takes the form of a normalised eigenvector corresponding to the eigenvalue closest to zero in absolute value. The iterative process can be started by computing the *algebraic least squares (ALS) estimate*,  $\hat{\boldsymbol{\theta}}_{\text{ALS}}$ , defined as the unconstrained minimiser of the cost function  $J_{\text{ALS}}(\boldsymbol{\theta}; \mathbf{x}_1, \dots, \mathbf{x}_n) =$

$\|\boldsymbol{\theta}\|^{-2} \sum_{i=1}^n \boldsymbol{\theta}^T \mathbf{A}_i \boldsymbol{\theta}$ , with  $\|\boldsymbol{\theta}\| = (\sum_{j=1}^6 \theta_j^2)^{1/2}$ . The estimate  $\widehat{\boldsymbol{\theta}}_{\text{ALS}}$  coincides, up to scale, with an eigenvector of  $\sum_{i=1}^n \mathbf{A}_i$  for the smallest eigenvalue, and this can be found via singular value decomposition as the right singular vector of the matrix  $[\mathbf{u}(x_1), \dots, \mathbf{u}(x_n)]^T$  corresponding to the smallest singular value.

With  $\mathbf{M}_\theta = \sum_{i=1}^n (\boldsymbol{\theta}^T \mathbf{B}_i \boldsymbol{\theta})^{-1} \mathbf{A}_i$  and  $\mathbf{N}_\theta = \sum_{i=1}^n (\boldsymbol{\theta}^T \mathbf{A}_i \boldsymbol{\theta})(\boldsymbol{\theta}^T \mathbf{B}_i \boldsymbol{\theta})^{-2} \mathbf{B}_i$ , equation (5) can equivalently be restated as

$$\mathbf{M}_\theta \boldsymbol{\theta} = \mathbf{N}_\theta \boldsymbol{\theta}, \quad (6)$$

where the evaluation at  $\widehat{\boldsymbol{\theta}}_{\text{AML}}^u$  is dropped for clarity. The *heteroscedastic errors-in-variables scheme* in its *basic form*, or *HEIV with intercept* [17–19], is based upon the observation that a vector  $\boldsymbol{\theta}$  satisfies (6) if and only if it is a solution of the *generalised* eigenvalue problem

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

corresponding to the eigenvalue  $\lambda = 1$ . Given a current approximate solution  $\boldsymbol{\theta}_c$ , an updated solution  $\boldsymbol{\theta}_+$  is taken to be a normalised eigenvector of the eigenvalue problem (7) corresponding to the eigenvalue closest to zero in absolute value. Again the iterative process can be seeded with  $\widehat{\boldsymbol{\theta}}_{\text{ALS}}$ .

## 5. CONSTRAINED MINIMISATION

A natural means for isolating the constrained minimiser  $\widehat{\boldsymbol{\theta}}_{\text{AML}}$  is the *constrained fundamental numerical scheme* (CFNS) [20]. The scheme is a variant of FNS in which  $\mathbf{X}_\theta$  is replaced by a more complicated symmetric matrix. As it turns out, CFNS is sensitive to the choice of the underlying coordinate system and its practical success depends critically on good pre-conditioning. This is so because not only the initial estimate has to be sufficiently close to the sought-after solution (as is the case with all Newton-like methods), but also the smallest eigenvalue of the counterpart of  $\mathbf{X}_\theta$  used in iterations has to be well separated from the remaining eigenvalues. As a rule, to meet these conditions, a transformation of the data-related variables needs to be applied as a pre-process and a conformal transformation of the parameters-related variables has to follow in a post-process. Work on a suitable pre-conditioning for the case in question is in progress.

A more conventional approach to constrained minimisation relies on the adoption of Kanatani’s *adjustment procedure* [16]. It is an iterative process operating on the result of unconstrained minimisation,  $\widehat{\boldsymbol{\theta}}_{\text{AML}}^u$ , by means of the update rule

$$\begin{aligned} \boldsymbol{\theta}_+ = & \boldsymbol{\theta}_c - [\partial_{\boldsymbol{\theta}} \phi(\boldsymbol{\theta}_c) \boldsymbol{\Lambda}_{\boldsymbol{\theta}_c} \partial_{\boldsymbol{\theta}} \phi(\boldsymbol{\theta}_c)^T]^{-1} \\ & \times \phi(\boldsymbol{\theta}_c) \boldsymbol{\Lambda}_{\boldsymbol{\theta}_c} \partial_{\boldsymbol{\theta}} \phi(\boldsymbol{\theta}_c)^T. \end{aligned}$$

Here  $\boldsymbol{\Lambda}_\theta = \mathbf{Q}_\theta (\mathbf{X}_{\widehat{\boldsymbol{\theta}}_{\text{AML}}^u})^{-1} \mathbf{Q}_\theta$ , with the notation  $\mathbf{A}^-$  for the Moore–Penrose pseudo-inverse of  $\mathbf{A}$ ,  $\mathbf{Q}_\theta = \mathbf{I}_l - \|\boldsymbol{\theta}\|^{-2} \boldsymbol{\theta} \boldsymbol{\theta}^T$ , with  $\mathbf{I}_l$  the  $l \times l$  identity matrix and  $l$  the length of  $\boldsymbol{\theta}$ , here set to 6. The limit estimate obtained via this method can be viewed as an approximate constrained minimiser of  $J_{\text{AML}}$ .

In an effort to find an alternative to CFNS capable of computing the genuine constrained minimiser for the problem to hand, we have developed a novel method which is akin to Kanatani’s adjustment procedure. It exploits iteratively the *sequential quadratic programming* (SQP) steps expressed by

$$\begin{aligned} \boldsymbol{\theta}_+ = & \boldsymbol{\theta}_c - [\partial_{\boldsymbol{\theta}} \phi(\boldsymbol{\theta}_c) \mathbf{H}_{\boldsymbol{\theta}_c}^- \partial_{\boldsymbol{\theta}} \phi(\boldsymbol{\theta}_c)^T]^{-1} \\ & \times \frac{\kappa + 1}{2} \phi(\boldsymbol{\theta}_c) \mathbf{H}_{\boldsymbol{\theta}_c}^- \partial_{\boldsymbol{\theta}} \phi(\boldsymbol{\theta}_c)^T. \end{aligned}$$

Here—let us recall— $\kappa$  is the homogeneity index of  $\phi$  (equal to 3) and  $\mathbf{H}_\theta$  is the Hessian of  $J_{\text{AML}}$  at  $\boldsymbol{\theta}$ , given explicitly by  $\mathbf{H}_\theta = 2(\mathbf{X}_\theta - \mathbf{T}_\theta)$ , where

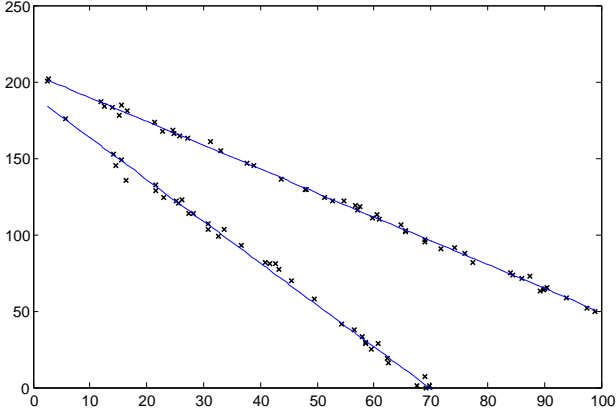
$$\begin{aligned} \mathbf{T}_\theta = & \sum_{i=1}^n \frac{2}{(\boldsymbol{\theta}^T \mathbf{B}_i \boldsymbol{\theta})^2} \left[ \mathbf{A}_i \boldsymbol{\theta} \boldsymbol{\theta}^T \mathbf{B}_i + \mathbf{B}_i \boldsymbol{\theta} \boldsymbol{\theta}^T \mathbf{A}_i \right. \\ & \left. - 2 \frac{\boldsymbol{\theta}^T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

When started close to  $\widehat{\boldsymbol{\theta}}_{\text{AML}}$  and repeated until convergence, the scheme delivers  $\widehat{\boldsymbol{\theta}}_{\text{AML}}$ . The SQP process can, but not necessarily must, be initialised with  $\widehat{\boldsymbol{\theta}}_{\text{AML}}^u$ . It is the lack of a tight pinning to the result of unconstrained minimisation that differentiates the new technique from Kanatani’s method.

It should be noted that while the value of the constraint residual  $|\phi|$  at successive updates generated by any iterative constrained minimisation technique like SQP systematically decreases as the computation progresses, the singularity constraint is hardly ever perfectly satisfied. The nearly perfect, but not ideal, satisfaction of the constraint means that, geometrically, the estimates are not pairs of lines, but are hyperbolae of high eccentricity—that is, hyperbolae that are elongated and have flat branches.

## 6. EVD CORRECTION

To ensure that two-line fitting algorithms produce usable estimates, a method is required for enforcing the perfect satisfaction of the singularity constraint. The method should be applicable to the final output of any two-line estimation procedure and, ideally, should deliver the result of the constraint enforcement in the form of a pair of lines. Here we describe one such correction technique based on EVD. It is tuned to fitting a pair of lines and does not directly generalise to fitting larger sets of lines. The method can be



**Fig. 1.** An example data set and corresponding (true) pair of lines.

viewed as an alternative to the factorisation technique proposed in [7].

A given estimate is first reshaped to take the form of a symmetric matrix  $C$ . Then EVD is performed on  $C$  yielding  $C = VDV^T$  with  $V = [v_1, v_2, v_3]$  orthogonal and  $D = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$ ,  $|\lambda_1| \geq |\lambda_2| \geq |\lambda_3|$ . Finally,  $C$  is modified to  $C_c = V D_c V^T$ , where  $D_c = \text{diag}(\lambda_1, \lambda_2, 0)$ . The corrected estimate  $C_c$  now perfectly satisfies the singularity constraint. This estimate can further be reinterpreted in accordance with the geometric nature of the associated set  $Z_{C_c}$ . If  $\lambda_1$  and  $\lambda_2$  are of opposite signs, then  $Z_{C_c}$  is the pair of lines

$$\begin{aligned} l_1 &= \sqrt{\text{sgn}(\lambda_1)\lambda_1}v_1 + \sqrt{\text{sgn}(\lambda_2)\lambda_2}v_2, \\ l_2 &= \sqrt{\text{sgn}(\lambda_1)\lambda_1}v_1 - \sqrt{\text{sgn}(\lambda_2)\lambda_2}v_2. \end{aligned} \quad (8)$$

If  $\lambda_2 = 0$ , then  $Z_{C_c}$  is the double line  $v_1$ . If  $\lambda_1$  and  $\lambda_2$  are of same sign, then  $Z_{C_c}$  represents the point  $v_3 = [v_{13}, v_{23}, v_{33}]^T$  in  $\mathbb{P}^2$ , which, when  $v_{33} \neq 0$ , belongs to  $\mathbb{R}^2$  and is given by  $[v_{13}/v_{33}, v_{23}/v_{33}, 1]^T$ . The last case can be viewed as exceptional and is not expected to arise frequently. In a typical situation, the input estimate  $C$  is such that the associated values  $\lambda_1$  and  $\lambda_2$  have opposite signs and the corrected estimate  $C_c$  is geometrically represented by the lines  $l_1$  and  $l_2$  given in (8).

## 7. EXPERIMENTS

To assess potential benefits stemming from the use of constrained optimisation in the realm of GPCA, we carried out a simulation study. Three algorithms, ALS, HEIV and SQP, were set to compute a pair of lines from synthetic data. We utilised a particular version of HEIV, namely the *reduced* HEIV scheme, or *HEIV without intercept*, that operates es-

Method	$J_{\text{AML}}$	Rank-2 $J_{\text{AML}}$
ALS	$1.581251 \times 10^2$	$2.396241 \times 10^4$
HEIV	3.814370	$1.494355 \times 10^4$
SQP	3.827415	3.827590

**Table 1.** Averages of testing results.

Method	$J_{\text{AML}}$	Rank-2 $J_{\text{AML}}$
ALS	1.164933	1.041400
HEIV	0.8503677	0.9816603
SQP	0.8952145	0.8952288

**Table 2.** Medians of testing results.

entially over a subspace of the parameter space of one dimension less [19]. The covariances of the data employed by HEIV and SQP were assumed to be the default  $2 \times 2$  identity matrix corresponding to isotropic homogeneous noise in image measurement.

To create data for our study, we randomly generated 10 pairs of lines. Along each line, in a section spanning 100 pixels in the  $x$  direction, 50 points were generated by sampling from a uniform distribution. To these points homogeneous zero-mean Gaussian noise was added with standard deviation of 1 pixel. This data was generated so as to represent the kinds of line segment that may be found by an edge detector. An example of the data is given in Figure 1.

Each estimator was applied to the points generated from each of the 10 pairs of lines and the resulting estimates were recorded and evaluated. As a measure of performance, we used the AML cost function, with the standard value of  $J_{\text{AML}}$  averaged across the points in the image.

To ensure that the outputs of the algorithms can be interpreted as genuine pairs of lines, all estimates were post-hoc EVD corrected. The  $J_{\text{AML}}$  value of the corrected estimates, Rank-2  $J_{\text{AML}}$ , is given in the rightmost columns in Tables 1 and 2. It is this Rank-2  $J_{\text{AML}}$  number that is the most informative indicator of the performance of a particular method.

Tables 1 and 2 give the cost function values for 3 types of estimates. Table 1 shows that, on average, HEIV is an effective minimiser of  $J_{\text{AML}}$ , and that SQP coupled with EVD correction produces better results than the EVD-corrected HEIV scheme. Moreover—and this is a critical observation—when applied to the SQP estimate, EVD correction leaves the  $J_{\text{AML}}$  value virtually unaffected (unlike in the case of the HEIV estimate, where EVD correction markedly worsens the  $J_{\text{AML}}$  value). This confirms that the result of constrained optimisation has an almost optimal form and that EVD correction in this case amounts to a tiny push, which can be stably executed in the presence of

noise.

Table 2 presents the results of the same tests but reports the median, rather than mean, of the  $J_{AML}$  values. As the median is usually more representative of the central tendency of a sample set than the mean, Table 2 provides a better indication of the performance of the algorithms on a typical trial.

## 8. CONCLUSIONS AND FUTURE WORK

We have presented a novel version of GPCA for the case of fitting a pair of lines to data, with a message extending to the general case of multi-component estimation. At the core of our formulation lies the reduction of the underlying estimation problem to minimisation of an error function having solid statistical foundations, subject to a parameter constraint. We have proposed a technique for isolating a constrained minimiser of that function. Preliminary experiments show that our algorithm provides better results than the standard GPCA based on unconstrained minimisation of the error function.

There are clearly a number of ways in which the work reported in this paper can be extended. The case of multiple lines can be approached starting from the observation that equation (1) characterising a pair of lines can equivalently be written as  $(\mathbf{l}_1 \otimes_s \mathbf{l}_2)^T (\mathbf{m} \otimes \mathbf{m}) = 0$ , where  $\mathbf{l}_1 \otimes_s \mathbf{l}_2 = (\mathbf{l}_1 \otimes \mathbf{l}_2 + \mathbf{l}_2 \otimes \mathbf{l}_1)/2$  is the symmetric tensor product of  $\mathbf{l}_1$  and  $\mathbf{l}_2$ , and  $\otimes$  denotes the Kronecker (or tensor) product. More generally, the equation for an aggregate of  $k$  lines is  $(\mathbf{l}_1 \otimes_s \cdots \otimes_s \mathbf{l}_k)^T (\mathbf{m} \otimes \cdots \otimes \mathbf{m}) = 0$ , where  $\mathbf{l}_1 \otimes_s \cdots \otimes_s \mathbf{l}_k = (k!)^{-1} \sum_{\sigma \in S_k} \mathbf{l}_{\sigma(1)} \otimes \cdots \otimes \mathbf{l}_{\sigma(k)}$  and  $S_k$  is the symmetric group on  $k$  elements. It is known that the totally decomposable symmetric tensors of the form  $\mathbf{l}_1 \otimes_s \cdots \otimes_s \mathbf{l}_k$  constitute an algebraic variety within the space of all symmetric tensors [21]. However, no explicit formula for the underlying constraints is known (this is a fundamental difference with the case of totally decomposable antisymmetric tensors). Working out these constraints in concrete cases like those involving low values of  $k$  will immediately allow the new version of GPCA to cope with larger multi-line structures. More generally, progress in applying the constrained GPCA to estimating more complicated multi-component structures will strongly depend on successful identification of relevant constraints.

## 9. REFERENCES

- [1] X.-M. Lou, L. G. Hassebrook, M. E. Lhamon, and J. Li, "Numerically efficient angle, width, offset, and discontinuity determination of straight lines by the discrete Fourier-bilinear transformation algorithm," *IEEE Trans. Image Processing*, vol. 6, no. 10, pp. 1464–1467, 1997.
- [2] V. Venkateswar and R. Chellappa, "Extraction of straight lines in aerial images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 11, pp. 1111–1114, 1992.
- [3] R. O. Duda and P. E. Hart, "Use of the Hough transform to detect lines and curves in pictures," *Commun. ACM*, vol. 15, pp. 11–15, 1972.
- [4] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*, Prentice Hall, 2003.
- [5] R. Vidal and Y. Ma, "A unified algebraic approach to 2-D and 3-D motion segmentation," in *Proc. Eighth European Conf. Computer Vision*, 2004, vol. 3021 of *Lecture Notes on Computer Vision*, pp. 1–15.
- [6] R. Vidal, Y. Ma, S. Soatto, and S. Sastry, "Two-view multibody structure from motion," *Int. J. Computer Vision*, to appear.
- [7] R. Vidal, Y. Ma, and S. Sastry, "General principal component analysis (GPCA)," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2003, vol. 1, pp. 621–628.
- [8] R. Vidal, Y. Ma, and J. Piazzi, "A new GPCA algorithm for clustering subspaces by fitting, differentiating and dividing polynomials," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2004, vol. 1, pp. 510–517.
- [9] A. Fitzgibbon, M. Pilu, and R. B. Fisher, "Direct least square fitting of ellipses," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 476–480, 1999.
- [10] R. Halif and J. Flusser, "Numerically stable direct least squares fitting of ellipses," in *Proc. Sixth Int. Conf. in Central Europe on Computer Graphics and Visualization*, 1998, pp. 125–132.
- [11] Y. Nievergelt, "Fitting conics of specific types to data," *Linear Algebra and Appl.*, vol. 378, pp. 1–30, 2004.
- [12] P. O'Leary and P. Zsombor-Murray, "Direct and specific least-square fitting of hyperbolæ and ellipses," *J. Electronic Imaging*, vol. 13, no. 3, pp. 492–503, 2004.
- [13] R. Horn and C. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, 1985.
- [14] M. J. Brooks, W. Chojnacki, D. Gawley, and A. van den Hengel, "What value covariance information in estimating vision parameters?," in *Proc. Eighth Int. Conf. Computer Vision*, 2001, vol. 1, pp. 302–308.
- [15] 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.*, vol. 22, no. 11, pp. 1294–1303, 2000.
- [16] K. Kanatani, *Statistical Optimization for Geometric Computation: Theory and Practice*, Elsevier, Amsterdam, 1996.
- [17] Y. Leedan and P. Meer, "Heteroscedastic regression in computer vision: Problems with bilinear constraint," *Int. J. Computer Vision*, vol. 37, no. 2, pp. 127–150, 2000.
- [18] B. Matei and P. Meer, "A general method for errors-in-variables problems in computer vision," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2000, vol. 2, pp. 18–25.
- [19] W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley, "From FNS and HEIV: A link between two vision parameter estimation methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 264–268, 2004.
- [20] 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*, vol. 22, pp. 85–91, 2004.
- [21] M. H. Lim, "Conditions on decomposable symmetric tensors as an algebraic variety," *Linear and Multilinear Algebra*, vol. 32, pp. 249–252, 1992.