Generalised Principal Component Analysis: Exploiting Inherent Parameter Constraints

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

School of Computer Science, University of Adelaide
Adelaide, SA 5005, Australia
`{wojciech.chojnacki,anton.vandenhengel,`
`michael.brooks}@adelaide.edu.au`

Abstract. Generalised Principal Component Analysis (GPCA) is a recently devised technique for fitting a multi-component, piecewise-linear structure to data that has found strong utility in computer vision. 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 a simple algebraic 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 a statistically motivated error function. The output of any GPCA algorithm hardly ever perfectly satisfies the parameter constraints. Our 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 example problem of fitting a pair of lines to noisy image points, but has potential for use in more general multi-component structure fitting in computer vision.

Keywords: Generalised principal component analysis, constrained minimisation, multi-line fitting, degenerate conic.

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 [Lou et al., 1997, Venkateswar and Chellappa, 1992]. Several methods have been proposed for solving this particular task, including those based on the Hough transform [Duda and Hart, 1972], K-subspaces [Ho et al., 2003], subspace growing and subspace selection [Leonardis et al., 2002], EM [Tipping and Bishop, 1999] and RANSAC [Forsyth and Ponce, 2003] algorithms. 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 [Vidal et al., 2002, Vidal and Ma, 2004, Vidal et al., 2006]. To tackle it, a new approach has been put forth under the label of *generalised principle component analysis* (GPCA) [Vidal et al., 2003, Vidal et al., 2004,

Vidal et al., 2005]. The GPCA method employs a parametric model in which parameters describe a multi-component, piecewise-linear structure to which various parts of a data set adhere. The number of linear components is assumed to be fixed and known beforehand. The relationship between data and components is encoded in a system of multivariate polynomial equations. When all components are hyperplanes, this system reduces to a single equation. In the special, but representative, case of fitting multiple lines to planar data, the order of the single polynomial describing the structure coincides with the number of the line 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 an approximate 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 [Vidal et al., 2003, Vidal et al., 2006] employs algebraic factorisation of a multivariate polynomial whose coefficients are obtained via *unconstrained* minimisation of a simple algebraic cost function, different from the one used in the present paper. The method does not require data segmentation and as such differs from iterative methods like K-subspaces and EM which alternate between estimating structure components and grouping the data around individual components. However, 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 [Vidal et al., 2004, Vidal and Ma, 2004, Vidal et al., 2005] 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 a statistically motivated cost function is minimised subject to appropriate constraints. 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 eigenvalue 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 [Fitzgibbon et al., 1999, Halíř and Flusser, 1998, Nievergelt, 2004, O'Leary and Zsombor-Murray, 2004].

2   Background

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

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

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

$$a m_1^2 + b m_1 m_2 + c m_2^2 + d m_1 + e m_2 + f = 0,$$

where a, b and c are not all zero. Introducing the symmetric matrix C

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

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

$$l_1^T m \cdot l_2^T m = m^T l_1 l_2^T m = 0$$

or equivalently, given that $m^T l_1 l_2^T m = m^T l_2 l_1^T m$,

$$m^T (l_1 l_2^T + l_2 l_1^T) m = 0. \tag{1}$$

With $C = l_1 l_2^T + l_2 l_1^T$, a symmetric matrix, the above equation can be rewritten as $m^T C m = 0$, showing that $Z_{l_1} \cup Z_{l_2}$ is identical with the conic Z_C. The matrices $l_i l_j^T$ are rank-1, so the rank of C is no greater than 2 and the conic is degenerate. If $l_1 = l_2$, then Z_C represents a single, repeated line; in this case the conic equation $(l_1^T m)^2 = 0$

is equivalent to the line equation $l_1^T m = 0$. 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

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

   To see that a pair of lines, a double line and a point are the only possible types of degenerate conic, suppose that C is a non-zero symmetric singular matrix. Then C admits an *eigenvalue decomposition* (EVD) of the form $C = VDV^T$, where V is an orthogonal 3×3 matrix and $D = \mathrm{diag}(\lambda_1, \lambda_2, \lambda_3)$, with λ_i $(i = 1, 2, 3)$ a real number [Horn and Johnson, 1985]. The eigenvalue decomposition differs from the *singular value decomposition* (SVD) of C in that the latter uses two orthogonal, possibly different, matrices U and V, and that the former uses a diagonal matrix whose entries are not necessarily non-negative. However, the EVD and SVD of the symmetric C are closely related—any of the two orthogonal factors U, V in the SVD can serve as V in the EVD, and 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 U and V differ by a sign, with all remaining entries left intact. For each $i = 1, 2, 3$, let v_i be the ith column vector of V. Then, clearly, v_i is an eigenvector of C corresponding to the eigenvalue λ_i, $Cv_i = \lambda_i v_i$, and, moreover, $C = \sum_{i=1}^{3} \lambda_i v_i v_i^T$. Now $\det C = \lambda_1 \lambda_2 \lambda_3 = 0$ so one eigenvalue, say λ_3, is zero, implying that $C = \sum_{i=1}^{2} \lambda_i v_i v_i^T$. If another eigenvalue, say λ_2, is zero too, then $C = \lambda_1 v_1 v_1^T$ and, since the remaining eigenvalue, λ_1, has to be non-zero, Z_C coincides with the line Z_{v_1}. If λ_3 is the only zero eigenvalue, then there are two possibilities—either λ_1 and λ_2 are of same sign, or λ_1 and λ_2 are of opposite sign. In the first case Z_C reduces to the linear span of $v_3 = [v_{13}, v_{23}, v_{33}]^T$ 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

$$\lambda_1 v_1 v_1^T + \lambda_2 v_2 v_2^T = l_1 l_2^T + l_2 l_1^T,$$

where $l_1 = \sqrt{\lambda_1} v_1 + \sqrt{-\lambda_2} v_2$ and $l_2 = \sqrt{\lambda_1} v_1 - \sqrt{-\lambda_2} v_2$. Consequently,

$$m^T C m = m^T l_1 l_2^T m + m^T l_2 l_1^T m = 2(l_1^T m)(l_2^T m),$$

so Z_C is the union of the lines Z_{l_1} and Z_{l_2}. The identification of Z_C with $Z_{l_1} \cup Z_{l_2}$ via the factorisation of the binomial $m^T C m$ as above exemplifies the general factorisation principle underlying GPCA.

3   Estimation Problem

The equation for a conic Z_C can alternatively be written as

$$\theta^T u(x) = 0, \tag{2}$$

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

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

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

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

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 $\boldsymbol{x}$ is the *ideal* datum representing the point.

Associated with this model 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 (2) associated with $\boldsymbol{x} = \boldsymbol{x}_i$ $(i = 1, \ldots, n)$, find $\boldsymbol{\theta} \neq \boldsymbol{0}$ satisfying (3) for which the cost function attains its minimum.

The 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*

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

where, for any length 2 vector $\boldsymbol{y}$, $\partial_{\boldsymbol{x}} \boldsymbol{u}(\boldsymbol{y})$ denotes the 6×2 matrix of the partial derivatives of the function $\boldsymbol{x} \mapsto \boldsymbol{u}(\boldsymbol{x})$ evaluated at $\boldsymbol{y}$, and, for each $i = 1, \ldots, n$, $\boldsymbol{\Lambda}_{\boldsymbol{x}_i}$ is a 2×2 symmetric *covariance matrix* describing the uncertainty of the data point $\boldsymbol{x}_i$ [Brooks et al., 2001, Chojnacki et al., 2000, Kanatani, 1996]. If J_{AML} is minimised over those non-zero parameter vectors for which (3) 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* $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}$. The unconstrained minimiser of J_{AML} obtained by ignoring the constraint (3) and searching over all of the parameter space defines the *unconstrained approximated likelihood estimate*, $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u$. The function $\boldsymbol{\theta} \mapsto J_{\mathrm{AML}}(\boldsymbol{\theta}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ is homogeneous of degree zero and the zero set of ϕ is invariant to multiplication by non-zero scalars, so both $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}$ and $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u$ are determined only up to scale. Obviously, $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}$ is the preferred estimate of $\boldsymbol{\theta}$, with $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u$ being the second best choice.

4 Unconstrained Minimisation

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

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

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

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

where

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

$$\boldsymbol{A}_i = \boldsymbol{u}(\boldsymbol{x}_i)\boldsymbol{u}(\boldsymbol{x}_i)^T,$$

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

The optimality condition rewritten as

$$[\boldsymbol{X}_{\boldsymbol{\theta}}\boldsymbol{\theta}]_{\theta=\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u} = \boldsymbol{0} \tag{5}$$

serves as the basis for isolating $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u$. Two Newton-like iterative algorithms can be used for solving (5). The *fundamental numerical scheme* (FNS) [Chojnacki et al., 2000] exploits the fact that a vector $\boldsymbol{\theta}$ satisfies (5) if and only if it is a solution of the *ordinary* eigenvalue problem

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

corresponding to the eigenvalue $\lambda = 0$. Given a current approximate solution $\boldsymbol{\theta}_c$, the stable version of FNS [Chojnacki et al., 2005] takes for an updated solution $\boldsymbol{\theta}_+$ a normalised eigenvector of $\boldsymbol{X}_{\boldsymbol{\theta}_c}$ corresponding to the smallest eigenvalue. The iterative 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 $J_{\mathrm{ALS}}(\boldsymbol{\theta}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \|\boldsymbol{\theta}\|^{-2}\sum_{i=1}^{n}\boldsymbol{\theta}^T\boldsymbol{A}_i\boldsymbol{\theta}$, with $\|\boldsymbol{\theta}\| = (\sum_{j=1}^{6}\theta_j^2)^{1/2}$. The estimate $\widehat{\boldsymbol{\theta}}_{\mathrm{ALS}}$ coincides, up to scale, with an eigenvector of $\sum_{i=1}^{n}\boldsymbol{A}_i$ for the smallest eigenvalue, and this can be found via singular value decomposition as the right singular vector of the matrix $[\boldsymbol{u}(\boldsymbol{x}_1), \ldots, \boldsymbol{u}(\boldsymbol{x}_n)]^T$ corresponding to the smallest singular value. Incidentally, we point out that the standard version of GCPA [Vidal et al., 2005] is based exclusively on ALS cost minimisation.

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

$$\boldsymbol{M}_{\boldsymbol{\theta}}\boldsymbol{\theta} = \boldsymbol{N}_{\boldsymbol{\theta}}\boldsymbol{\theta}, \tag{6}$$

where the evaluation at $\widehat{\boldsymbol{\theta}}_{\mathrm{AML}}^u$ is dropped for clarity. The *heteroscedastic errors-in-variables scheme* in its *basic* form, or *HEIV with intercept* [Leedan and Meer, 2000, Matei and Meer, 2000, Chojnacki et al., 2004a], 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

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

corresponding to the eigenvalue $\lambda = 1$. Given a current approximate solution $\boldsymbol{\theta}_c$, HEIV takes for an updated solution $\boldsymbol{\theta}_+$ a normalised eigenvector of the eigenvalue problem $\boldsymbol{M}_{\boldsymbol{\theta}_c}\boldsymbol{\xi} = \lambda\boldsymbol{N}_{\boldsymbol{\theta}_c}\boldsymbol{\xi}$ corresponding to the smallest eigenvalue. Again the iterative process can be seeded with $\widehat{\boldsymbol{\theta}}_{\mathrm{ALS}}$.

5   Approximate Constrained Minimisation

A natural means for isolating the constrained minimiser $\widehat{\boldsymbol{\theta}}_{\text{AML}}$ is the *constrained fundamental numerical scheme* (CFNS) [Chojnacki et al., 2004b]. The scheme is a variant of FNS in which $\boldsymbol{X}_{\boldsymbol{\theta}}$ 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 $\boldsymbol{X}_{\boldsymbol{\theta}}$ 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.

To find an estimate satisfying the singularity constraint and having the property that the value of J_{AML} at that estimate is only slightly increased compared to $J_{\text{AML}}(\widehat{\boldsymbol{\theta}}^u_{\text{AML}})$, we take a more conventional approach and adopt an *adjustment procedure*. It is a separate post-process operating on the result of unconstrained minimisation, $\widehat{\boldsymbol{\theta}}^u_{\text{AML}}$. The estimate obtained via a post-hoc correction can be viewed as an approximate constrained minimiser.

A standard adjustment technique, due to Kanatani [Kanatani, 1996], generates iteratively a sequence of estimates, starting from $\widehat{\boldsymbol{\theta}}^u_{\text{AML}}$, with the use of the update rule

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

Here $\boldsymbol{\Lambda}_{\boldsymbol{\theta}} = \boldsymbol{Q}_{\boldsymbol{\theta}}(\boldsymbol{X}_{\widehat{\boldsymbol{\theta}}^u_{\text{AML}}})^- \boldsymbol{Q}_{\boldsymbol{\theta}}$, with the notation $\boldsymbol{A}^-$ for the Moore-Penrose pseudo-inverse of $\boldsymbol{A}$, $\boldsymbol{Q}_{\boldsymbol{\theta}} = \boldsymbol{I}_l - \|\boldsymbol{\theta}\|^{-2}\boldsymbol{\theta}\boldsymbol{\theta}^T$, with $\boldsymbol{I}_l$ the $l \times l$ identity matrix and l the length of $\boldsymbol{\theta}$, here set to 6. The scheme is repeated until the value of the constraint residual $|\phi|$ is acceptably small. The final estimate delivers an approximation to $\widehat{\boldsymbol{\theta}}_{\text{AML}}$.

In an effort to achieve a greater resemblance to CFNS, we have developed an alternative post-hoc correction (PHC) technique. It exploits the iterative process

$$\boldsymbol{\theta}_+ = \boldsymbol{\theta}_c - [\partial_{\boldsymbol{\theta}}\phi(\boldsymbol{\theta}_c)\boldsymbol{H}^-_{\boldsymbol{\theta}_c}\partial_{\boldsymbol{\theta}}\phi(\boldsymbol{\theta}_c)^T]^{-1} \times \phi(\boldsymbol{\theta}_c)\boldsymbol{H}^-_{\boldsymbol{\theta}_c}\partial_{\boldsymbol{\theta}}\phi(\boldsymbol{\theta}_c)^T.$$

Here $\boldsymbol{H}_{\boldsymbol{\theta}}$ is the Hessian of J_{AML} at $\boldsymbol{\theta}$, given explicitly by $\boldsymbol{H}_{\boldsymbol{\theta}} = 2(\boldsymbol{X}_{\boldsymbol{\theta}} - \boldsymbol{T}_{\boldsymbol{\theta}})$, where

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

As in Kanatani's method, the process is initialised with $\widehat{\boldsymbol{\theta}}_{\text{AML}}$ and is continued until the value of the constraint residual is sufficiently small.

It should be noted that while the value of the constraint residual at successive updates generated by any iterative (approximate) constrained minimisation technique like PHC 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 singularity constraint in a perfect manner. 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 viewed as an alternative to the factorisation technique proposed in [Vidal et al., 2003].

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 $D = \mathrm{diag}(\lambda_1, \lambda_2, \lambda_3)$, $|\lambda_1| \geq |\lambda_2| \geq |\lambda_3|$. Finally, C is modified to $C_c = VD_cV^T$, where $D_c = \mathrm{diag}(\lambda_1, \lambda_2, 0)$. 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 λ_1 and λ_2 are of opposite signs, then Z_{C_c} is the pair of lines

$$
\begin{aligned}
l_1 &= \sqrt{\mathrm{sgn}(\lambda_1)\lambda_1}\,v_1 + \sqrt{\mathrm{sgn}(\lambda_2)\lambda_2}\,v_2, \\
l_2 &= \sqrt{\mathrm{sgn}(\lambda_1)\lambda_1}\,v_1 - \sqrt{\mathrm{sgn}(\lambda_2)\lambda_2}\,v_2.
\end{aligned}
\tag{7}
$$

If $\lambda_2 = 0$, then Z_{C_c} is the double line v_1. If λ_1 and λ_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 λ_1 and λ_2 have opposite signs and the corrected estimate C_c is geometrically represented by the lines l_1 and l_2 given in (7).

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 PHC (described in Sections 4 and 5), 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 essentially over a subspace of the parameter space of one dimension less [Chojnacki et al., 2004a]. The covariances of the data employed by HEIV and PHC were assumed to be the default 2×2 identity matrix corresponding to isotropic homogeneous noise in image measurement.

To create data for our study, we randomly generated 100 pairs of lines. Along each line, in a section spanning 100 pixels in the x direction, 100 points were generated by sampling from a uniform distribution. To these points homogeneous zero-mean Gaussian noise was added at three different levels characterised by the standard deviation σ of 0.1, 0.55 and 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 100 pairs of lines and the resulting estimates were recorded and evaluated. As a measure of performance,
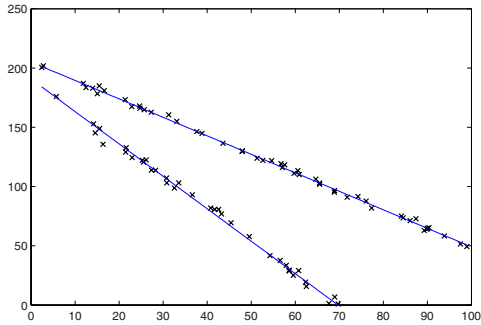
Fig. 1. An example data set and corresponding (true) pair of lines

Table 1. Averages of testing results

σ	Method	J_{AML}	Rank-2 J_{AML}
0.1	ALS	1.286×10^{-1}	2.060×10^{1}
	HEIV	1.084×10^{-2}	8.421×10^{-2}
	PHC	1.065×10^{-2}	1.065×10^{-2}
0.55	ALS	1.444×10^{1}	5.977×10^{5}
	HEIV	4.098	1.801×10^{2}
	PHC	9.190	9.195
1.0	ALS	2.779×10^{2}	5.448×10^{2}
	HEIV	4.448	3.652×10^{1}
	PHC	2.816	2.816

we used the AML cost function, with the standard value of J_{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_{AML} value of the corrected estimates, Rank-2 J_{AML}, is given in the rightmost columns in Tables 1 and 2. It is this Rank-2 J_{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_{AML}, and that PHC coupled with EVD correction produces better results that the EVD-corrected HEIV scheme. Moreover—and this is a critical observation—when applied to the PHC estimate, EVD correction leaves the J_{AML} value virtually unaffected (unlike in the case of the HEIV estimate, where EVD correction markedly worsens the J_{AML} value). This confirms that the result of approximate 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. Medians of testing results

σ	Method	J_{AML}	Rank-2 J_{AML}
	ALS	1.135×10^{-2}	1.113×10^{-2}
0.1	HEIV	9.517×10^{-3}	1.037×10^{-2}
	PHC	9.713×10^{-3}	9.713×10^{-3}
	ALS	3.738×10^{-1}	3.687×10^{-1}
0.55	HEIV	3.092×10^{-1}	3.277×10^{-1}
	PHC	3.121×10^{-1}	3.121×10^{-1}
	ALS	1.094	1.212
1.0	HEIV	9.498×10^{-1}	1.093
	PHC	9.509×10^{-1}	9.509×10^{-1}

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 an approximate constrained minimiser of that function. Preliminary experiments show that our algorithm provides better results than the standard GPCA based on unconstrained minimisation of a simple algebraic 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 $(l_1 \otimes_s l_2)^T (m \otimes m) = 0$, where $l_1 \otimes_s l_2 = (l_1 \otimes l_2 + l_2 \otimes l_1)/2$ is the symmetric tensor product of l_1 and l_2, and $\otimes$ denotes the Kronecker (or tensor) product. More generally, the equation for an aggregate of k lines is $(l_1 \otimes_s \cdots \otimes_s l_k)^T (m \otimes \cdots \otimes m) = 0$, where $l_1 \otimes_s \cdots \otimes_s l_k = (k!)^{-1} \sum_{\sigma \in S_k} l_{\sigma(1)} \otimes \cdots \otimes l_{\sigma(k)}$ and S_k is the symmetric group on k elements. It is known that the totally decomposable symmetric tensors of the form $l_1 \otimes_s \cdots \otimes_s l_k$ constitute an algebraic variety within the space of all symmetric tensors [Lim, 1992]. 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.

References

Brooks, M.J., Chojnacki, W., Gawley, D., van den Hengel, A.: What value covariance information in estimating vision parameters? In: Proc. Eighth Int. Conf. Computer Vision, vol. 1, pp. 302–308 (2001)

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

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

Chojnacki, W., Brooks, M.J., van den Hengel, A., Gawley, D.: A new constrained parameter estimator for computer vision applications. Image and Vision Computing 22, 85–91 (2004b)

Chojnacki, W., Brooks, M.J., van den Hengel, A., Gawley, D.: FNS, CFNS and HEIV: A unifying approach. J. Math. Imaging and Vision 23(2), 175–183 (2005)

Duda, R.O., Hart, P.E.: Use of the Hough transform to detect lines and curves in pictures. Commun. ACM 15, 11–15 (1972)

Fitzgibbon, A., Pilu, M., Fisher, R.B.: Direct least square fitting of ellipses. IEEE Trans. Pattern Anal. Mach. Intell. 21(5), 476–480 (1999)

Forsyth, D.A., Ponce, J.: Computer Vision: A Modern Approach. Prentice-Hall, Englewood Cliffs (2003)

Halíř., Flusser, J.: Numerically stable direct least squares fitting of ellipses. In: Proc. Sixth Int. Conf. in Central Europe on Computer Graphics and Visualization, pp. 125–132 (1998)

Ho, J., Yang, M.-H., Lim, J., Lee, K.-C., Kriegman, D.J.: Clustering appearances of objects under varying illumination conditions. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 11–18. IEEE, Los Alamitos (2003)

Horn, R., Johnson, C.: Matrix Analysis. Cambridge University Press, Cambridge (1985)

Kanatani, K.: Statistical Optimization for Geometric Computation: Theory and Practice. Elsevier, Amsterdam (1996)

Leedan, Y., Meer, P.: Heteroscedastic regression in computer vision: Problems with bilinear constraint. Int. J. Computer Vision 37(2), 127–150 (2000)

Leonardis, A., Bischof, H., Maver, J.: Multiple eigenspaces. Pattern Recognition 35(11), 2613–2627 (2002)

Lim, M.H.: Conditions on decomposable symmetric tensors as an algebraic variety. Linear and Multilinear Algebra 32, 249–252 (1992)

Lou, X.-M., Hassebrook, L.G., Lhamon, M.E., Li, J.: Numerically efficient angle, width, offset, and discontinuity determination of straight lines by the discrete Fourier-bilinear transformation algorithm. IEEE Trans. Image Processing 6(10), 1464–1467 (1997)

Matei, B., Meer, P.: A general method for errors-in-variables problems in computer vision. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 2, pp. 18–25. IEEE, Los Alamitos (2000)

Nievergelt, Y.: Fitting conics of specific types to data. Linear Algebra and Appl. 378, 1–30 (2004)

O'Leary, P., Zsombor-Murray, P.: Direct and specific least-square fitting of hyperbolæ and ellipses. J. Electronic Imaging 13(3), 492–503 (2004)

Tipping, M.E., Bishop, C.M.: Mixtures of probabilistic principal component analysers. Neural Computation 11(2), 443–482 (1999)

Venkateswar, V., Chellappa, R.: Extraction of straight lines in aerial images. IEEE Trans. Pattern Anal. Mach. Intell. 14(11), 1111–1114 (1992)

Vidal, R., Ma, Y.: A unified algebraic approach to 2-D and 3-D motion segmentation. In: Pajdla, T., Matas, J. (eds.) ECCV 2004. LNCS, vol. 3021, pp. 1–15. Springer, Heidelberg (2004)

Vidal, R., Ma, Y., Piazzi, J.: A new GPCA algorithm for clustering subspaces by fitting, differentiating and dividing polynomials. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 510–517. IEEE, Los Alamitos (2004)

Vidal, R., Ma, Y., Sastry, S.: Generalized principal component analysis (GPCA). In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 621–628. IEEE, Los Alamitos (2003)

Vidal, R., Ma, Y., Sastry, S.: Generalized principal component analysis (GPCA). IEEE Trans. Pattern Anal. Mach. Intell. 27(12), 1945–1959 (2005)

Vidal, R., Ma, Y., Soatto, S., Sastry, S.: Segmentation of dynamical scenes from the multibody fundamental matrix. In: ECCV Workshop Vision and Modelling of Dynamical Scenes (2002) Available at `http://www.robots.ox.ac.uk/~awf/eccv02/vamods02-rvidal.pdf`

Vidal, R., Ma, Y., Soatto, S., Sastry, S.: Two-view multibody structure from motion. Int. J. Computer Vision 68(1), 7–25 (2006)