# GENERALIZED PRINCIPAL COMPONENT ANALYSIS (GPCA)

by

Rene Vidal, Yi Ma and Shankar Sastry

Memorandum No. UCB/ERL M02/15

14 May 2002

# GENERALIZED PRINCIPAL COMPONENT ANALYSIS (GPCA)

by

Rene Vidal, Yi Ma and Shankar Sastry

Memorandum No. UCB/ERL M02/15

14 May 2002

**ELECTRONICS RESEARCH LABORATORY**

College of Engineering
University of California, Berkeley
94720

# Generalized Principal Component Analysis (GPCA)

Rene Vidal *
*Electrical Engineering and Computer Sciences*
*University of California at Berkeley*
*301 Cory Hall, Berkeley, CA 94720*
*Tel: (510)-643-2382* *E-mail:* `rvidal@eecs.berkeley.edu`

Yi Ma
*Electrical & Computer Engineering Department*
*University of Illinois at Urbana-Champaign*
*1406 West Green Street, Urbana, IL 61801*
*Tel: (217)-244-0871* *E-mail:* `yima@uiuc.edu`

Shankar Sastry
*Department of Electrical Engineering and Computer Sciences*
*University of California at Berkeley*
*237 Cory Hall, Berkeley, CA 94720*
*Tel: (510)-642-1857* *E-mail:* `sas ry@eecs.berkeley.edu`

May 14, 2002

**Abstract.** In this paper, we consider the so-called *Generalized Principal Component Analysis* (GPCA) problem, i.e. the problem of identifying $n$ linear subspaces of a $K$-dimensional linear space from a collection of sample points drawn from these subspaces. We cast the GPCA problem in an algebraic geometric framework and show that it is essentially equivalent to a factorization problem in the space of homogeneous polynomials of degree $n$ in $K$ variables. We prove that such a problem has a unique solution which can be obtained from the roots of a polynomial of degree $n$ in one variable and from the solution of $K-2$ linear systems in $n$ variables. Therefore, the GPCA problem has a closed form solution when $n \leq 4$. Furthermore, we show that the number of subspaces $n$ can also be obtained from the rank of a certain matrix that depends on the data. The theory of GPCA presented in this paper can be applied to a variety of estimation problems in which the data comes simultaneously from multiple (approximately) linear models. In this paper we apply GPCA to the estimation of a mixture of probabilistic models without any knowledge about the distribution of the data. We also apply GPCA to the multibody structure from motion problem in computer vision, i.e. the problem of estimating the 3D motion of multiple moving objects from 2D imagery. Applications to image grouping, handwritten digit recognition, texture segmentation and face recognition are forthcoming.

**Keywords:** Principal component analysis, factorization of homogeneous polynomials, mixture of probabilistic models, multibody structure from motion.

* Research supported by ONR grant N00014-00-1-0621 and UIUC ECE Department startup fund.

## 1. Introduction

Principal Component Analysis (PCA) refers to the problem of identifying a linear subspace $S \subset \mathbb{R}^K$ of unknown dimension $k < K$ from $N$ sample points $x^j \in S$, $j = 1, \ldots, N$. This problem shows up in a variety of applications in many fields, e.g., face and object recognition [5], handwritten digit recognition,dynamic textures [4], etc., just to mention a few. In spite of its enormous applicability, PCA can be solved in a remarkably simple way from the singular value decomposition (SVD) of the data matrix $[x^1 \cdots x^N] \in \mathbb{R}^{K \times N}$.

A natural generalization of PCA is to consider a mixture of principal components, in which the sample points $\{x^j \in \mathbb{R}^K\}_{j=1}^N$ are drawn from $n > 1$ linear subspaces of $\mathbb{R}^K$, $\{S_i\}_{i=1}^n$, as illustrated in Figure 1 for $n=3$ and $K=3$. In this case, the problem becomes that of identifying each subspace without knowing which sample points belong to which subspace[1]. It is natural to ask if this problem can be solved with some generalization of the SVD. It turns out that even though SVD has a multi-linear counterpart, the so-called higher order singular value decomposition (HOSVD) [2], such a generalization is not unique. Furthermore, while the SVD of a matrix $A = U\Sigma V^T$ produces a diagonal matrix $\Sigma$, the HOSVD of a tensor $\mathcal{A}$ produces a tensor $\mathcal{S}$ which is hardly diagonal. Thus, it is not possible to apply HOSVD to the mixture of PCAs problem.

A traditional approach to mixtures of principal components, usually referred to as *Probabilistic PCA* (PPCA), assumes that sample points are drawn from a certain probability distribution. The parameters of the distribution are estimated in a Maximum Likelihood or Maximum a Posteriori framework as follows: one first estimates the likelihood of the mixture of models given a prior on the grouping of the data, and then estimates the likelihood of the grouping given the current estimation of the subspaces. This is usually done in an iterative manner using the Expectation Maximization (EM) algorithm.

In our opinion, the probabilistic approach to mixtures of principal components suffers from the following disadvantages:

1. It relies on a probabilistic model for the data, which is restricted to certain classes of distributions or independence assumptions.

2. The EM algorithm depends on initialization. In fact, to the best of our knowledge, there is no global initialization irrespective of the distribution of the data. Furthermore, there is no guarantee that the EM algorithm will converge to the optimal solution.

[1] If the association between points and subspaces is know, then the problem reduces to standard PCA applied to each subspace.

3. It is hard to analyze some theoretical questions such as the existence and uniqueness of a solution to the problem.

4. There are many cases in which it is very hard to solve the grouping problem correctly, and yet it is possible to obtain a quite precise estimate of the subspaces. In those cases, a direct estimation of the subspaces (without grouping) seems more appropriate than an estimation based on incorrectly segmented data.

In this paper, we propose a novel approach the so-called *Generalized Principal Component Analysis (GPCA)* problem, which under mild assumptions guarantees a unique global solution based on simple linear algebraic techniques. We assume no probabilistic model for the data. Instead, we cast the GPCA problem in an algebraic geometric framework and show that it is essentially equivalent to a factorization problem in the space of homogeneous polynomials of degree $n$ in $K$ variables. We prove that such a problem has a unique solution which can be obtained from the roots of a polynomial of degree $n$ in one variable and from the solution of $K - 2$ linear systems in $n$ variables. Therefore, the GPCA problem has a closed form solution when $n \leq 4$. Furthermore, we show that the number of subspaces $n$ can also be obtained from the rank of a certain matrix that depends on the data.

The theory of GPCA presented in this paper can be applied to a variety of estimation problems in which the data comes simultaneously from multiple (approximately) linear models. In this paper we apply GPCA to the estimation of a mixture of probabilistic models without any knowledge about the distribution of the data. We also apply GPCA to the multibody structure from motion problem in computer vision, i.e. the problem of estimating the 3D motion of multiple moving objects from 2D imagery. Applications to image grouping, handwritten digit recognition, texture segmentation and face recognition are forthcoming.

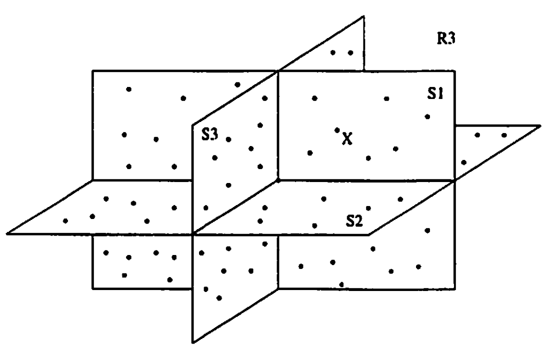



*Figure 1.* Three ($n = 3$) 2-dimensional subspaces $S_1, S_2, S_3$ in $\mathbb{R}^3$. GPCA tries to identify all three subspaces from samples $\{x\}$ drawn from these subspaces.

## 2. Generalized Principal Component Analysis (GPCA)

In this paper, we consider the following generalization of PCA:

PROBLEM 1 (Generalized Principal Component Analysis (GPCA)). *Given a set of sample points $\{x^j \in \mathbb{R}^K\}_{j=1}^N$ drawn from $n > 1$ distinct linear subspaces $\{S_i \subset \mathbb{R}^K\}_{i=1}^n$ of dimension $K-1$, $i = 1, \ldots, n$, identify each subspace $S_i$ without knowing which sample points belong to which subspace. By identifying the subspaces we mean the following:*

1. *Identify the number of subspaces $n$;*
2. *Identify a basis (or a set of principal components) for each subspace $S_i$ (or equivalently $S_i^\perp$);*
3. *Group or segment the $N$ points into the subspace(s) they belong to;*
4. *Determine the conditions under which 1-3 can be done, and if there is a unique closed-form solution to the problem.*

Each subspace $S_i$ is a $(K-1)$-dimensional space in $R^K$ that can be defined in terms of a nonzero *normal* vector $b_i \in \mathbb{R}^K$ as follows:

$$S_i = \{x \in \mathbb{R}^K : b_i^T x = b_{i1}x_1 + b_{i2}x_2 + \ldots + b_{iK}x_K = 0\}. \tag{1}$$

Since the subspaces $S_i$ are all distinct from each other, we assume that the normal vectors $\{b_i\}_{i=1}^n$ are pairwise linearly independent.

Now imagine that we are given a point $x \in \mathbb{R}^K$ lying on one of the subspaces $S_i$. Such a point must satisfy the formula:

$$(b_1^T x) = 0 \vee (b_2^T x) = 0 \vee \cdots \vee (b_n^T x) = 0, \tag{2}$$

which is equivalent to the following homogeneous polynomial of degree $n$ in $x$ with real coefficients:

$$\boxed{p_n(x) = \prod_{i=1}^n (b_i^T x) = 0.} \tag{3}$$

The problem of identifying each subspace $S_i$ is then equivalent to that of solving for the vectors $b_i$'s from the *nonlinear* equation (3). A standard technique used in algebra to render a nonlinear problem into a linear one is to find an "embedding" that lifts the problem into a higher-dimensional space. Let $R[x_1, \ldots, x_K]$ or simply $R(K)$ be the ring of all polynomials with real coefficients in $K$ variables. Its subset of all homogeneous polynomials of degree $n$ can be denoted as

$R_n(K)$. Therefore the whole ring $R(K)$ is a direct sum of homogeneous polynomials of different degrees[2]

$$R(K) = R_0(K) \oplus R_1(K) \oplus \cdots \oplus R_n(K) \oplus \cdots .$$

We notice that each $R_n(K)$, can be made into a vector space under the usual addition and scalar multiplication. Furthermore, $R_n(K)$ is generated by the set of monomials $x_1^{n1} x_2^{n2} \cdots x_K^{n_K}$, with $0 \leq n_j \leq n$, $j = 1, \ldots, K$, and $n_1 + n_2 + \cdots + n_K = n$. It is readily seen that there are a total of

$$M_n = \binom{n+K-1}{K-1} = \binom{n+K-1}{n} \tag{4}$$

different monomials, thus the dimension of $R_n(K)$ as a vector space is $M_n$. Therefore, we can define the following embedding (or lifting) from $\mathbb{R}^K$ into $\mathbb{R}^{M_n}$:

DEFINITION 1 (Veronese map). *Given $n$ and $K$, the* Veronese map *of degree $n$, $\nu_n : \mathbb{R}^K \to \mathbb{R}^{M_n}$, is defined as:*

$$\nu_n : \; [x_1, \ldots, x_K]^T \mapsto [\ldots, x^I, \ldots]^T, \tag{5}$$

*where $x^I$ is a monomial of the form $x_1^{n1} x_2^{n2} \cdots x_K^{n_K}$ with $I$ chosen in the degree-lexicographic order.*

EXAMPLE 1. *If $\boldsymbol{x} \in \mathbb{R}^2$, the Veronese map of degree $n$ is given by:*

$$\nu_n(x_1, x_2) = [x_1^n, x_1^{n-1}x_2, x_1^{n-2}x_2^2, \ldots, x_2^n]^T \tag{6}$$

With the so-defined Veronese map, equation (3) becomes the following linear expression in $\boldsymbol{a} \in \mathbb{R}^{M_n}$:

$$p_n(\boldsymbol{x}) = \nu_n(\boldsymbol{x})^T \boldsymbol{a} = \sum a_{n_1, n_2, \ldots, n_K} x_1^{n_1} x_2^{n_2} \cdots x_K^{n_K} = 0 \tag{7}$$

where $a_I \in \mathbb{R}$ represents the coefficient of monomial $x^I$. Notice that each $a_I$ is a symmetric multilinear function of $(\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n)$, that is $a_I$ is linear in each $\boldsymbol{b}_i$ and:

$$a_I(\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n) = a_I(\boldsymbol{b}_{\sigma(1)}, \boldsymbol{b}_{\sigma(2)}, \ldots, \boldsymbol{b}_{\sigma(n)}) \text{ for all } \sigma \in \mathfrak{S}_n, \tag{8}$$

where $\mathfrak{S}_n$ is the permutation group of $n$ elements.

[2] In Algebra, $R(K)$ is also called a graded ring of polynomials graded by degree.

EXAMPLE 2. *If $n=2$ and $K=3$, then we have*

$$p_2(\boldsymbol{x}) = (b_{11}x_2 + b_{12}x_2 + b_{13}x_3)(b_{21}x_1 + b_{22}x_2 + b_{23}x_3)$$
$$\nu_2(\boldsymbol{x}) = [x_1^2, x_1x_2, x_1x_3, x_2^2, x_2x_3, x_3^2]^T$$
$$\boldsymbol{a} = [\underbrace{b_{11}b_{21}}_{a_{2,0,0}}, \underbrace{b_{11}b_{22}+b_{12}b_{21}}_{a_{1,1,0}}, \underbrace{b_{11}b_{23}+b_{13}b_{21}}_{a_{1,0,1}}, \underbrace{b_{12}b_{22}}_{a_{0,2,0}}, \underbrace{b_{12}b_{23}+b_{13}b_{22}}_{a_{0,1,1}}, \underbrace{b_{13}b_{23}}_{a_{0,0,2}}]^T.$$

REMARK 1 (Symmetric Tensors). *Any homogeneous polynomial of degree $n$ in $K$ variables is also symmetric $n$-th order tensors in $K$ variables. Furthermore, the coefficients, $\boldsymbol{a}$, of polynomial $p_n(\boldsymbol{x})$ can be interpreted as the symmetric tensor product of the coefficients $\boldsymbol{b}_i$'s of each polynomial of degree 1, that is:*

$$\boldsymbol{a} \simeq Sym(\boldsymbol{b}_1 \otimes \boldsymbol{b}_2 \otimes \ldots \otimes \boldsymbol{b}_n) = \sum_{\sigma \in \mathfrak{S}_n} \boldsymbol{b}_{\sigma(1)} \otimes \boldsymbol{b}_{\sigma(2)} \otimes \cdots \otimes \boldsymbol{b}_{\sigma(n)}$$

*where $\otimes$ represents the tensor of Kronecker product and $\simeq$ represents the homemorphism between the symmetric tensor $Sym(\boldsymbol{b}_1 \otimes \boldsymbol{b}_2 \otimes \ldots \otimes \boldsymbol{b}_n)$ in $Sym^n(\mathbb{R}^K)$ and its symmetric part written as a vector $\boldsymbol{a}$ in $\mathbb{R}^{M_n}$.*

### 2.1. Estimation of the number of subspaces $n$

Given a collection of $N \geq M_n$ sample points $\{\boldsymbol{x}^j\}_{j=1}^N$, the vector of coefficients $\boldsymbol{a}$ satisfies the system of linear equations:

$$L_n\, \boldsymbol{a} = \begin{bmatrix} \nu_n(\boldsymbol{x}^1)^T \\ \nu_n(\boldsymbol{x}^2)^T \\ \vdots \\ \nu_n(\boldsymbol{x}^N)^T \end{bmatrix} \boldsymbol{a} = 0 \in \mathbb{R}^N. \tag{9}$$

We are now interested in determining whether there exist a unique solution for $\boldsymbol{a}$ (up to scale) from system (9), i.e. we would like to know under what conditions we have $\operatorname{rank}(L_n) = M_n - 1$. It turns out that the uniqueness of $\boldsymbol{a}$ is very much related to the estimation of the number of subspaces $n$ as shown by the following proposition and its corollary:

PROPOSITION 1 (Number of subspaces). *Assume that a collection of $N$ sample points $\{\boldsymbol{x}^j\}_{j=1}^N$ on $n$ different $(K-1)$-dimensional subspaces in $\mathbb{R}^K$ is given. Consider the Veronese map $\nu_i(\boldsymbol{x})$ of degree $i$, and let $L_i \in \mathbb{R}^{N \times M_i}$ be the matrix defined in (9). If the number of sample* *points in each subspace is big enough and the sample points are in general position, we have:*

$$rank(L_i) \begin{cases} > M_i - 1, & i < n, \\ = M_i - 1, & i = n, \\ < M_i - 1, & i > n. \end{cases} \tag{10}$$

*Therefore, the number n of subspaces is given by:*

$$n = min\{i : rank(L_i) = M_i - 1\}. \tag{11}$$

*Proof.* Consider the polynomial $p_n(\boldsymbol{x})$ as a polynomial over the algebraically closed field $\mathbb{C}$ and assume that each plane $\boldsymbol{b}_i^T\boldsymbol{x} = 0$ is different from each other. Then the ideal $I$ generated by $p_n(\boldsymbol{x})$ is a *radical ideal* with $p_n(\boldsymbol{x})$ as its only generator. According to Hilbert's Nullstellensatz (see page 380, [3]),there is a one-to-one correspondence between such an ideal $I$ and the algebraic set (also called algebraic variety in Algebra)

$$Z(I) \doteq \{\boldsymbol{x} : \forall p \in I, p(\boldsymbol{x}) = 0\} \subset \mathbb{C}^K$$

associated to it. Hence its generator $p_n(\boldsymbol{x})$ is uniquely determined by points in this algebraic set. By definition, $p_n(\boldsymbol{x})$ has the lowest degree among all the elements in the ideal $I$. Hence no polynomial with lower degree would vanish on all points in these subspaces. Furthermore, since all coefficients $\boldsymbol{b}_i$ are real, if $\boldsymbol{x} + \sqrt{-1}\boldsymbol{y} \in \mathbb{C}^K$ is in $Z(I)$, both $\boldsymbol{x} \in \mathbb{R}^K$ and $\boldsymbol{y} \in \mathbb{R}^K$ are in the set of (real) subspaces, because $\boldsymbol{b}_i^T(\boldsymbol{x}+\sqrt{-1}\boldsymbol{y}) = 0 \Leftrightarrow \boldsymbol{b}_i^T\boldsymbol{x} = 0$ & $\boldsymbol{b}_i^T\boldsymbol{y} = 0$. Hence all points on the (real) subspaces determine the polynomial $p_n(\boldsymbol{x})$ uniquely and vice-versa. ∎

COROLLARY 1. *The vector of coefficients $\boldsymbol{a}$ of the homogeneous polynomial $p_n(\boldsymbol{x})$ can be uniquely determined as the kernel of the matrix $L_n \in \mathbb{R}^{N \times M_n}$ from at least $M_n - 1$ points $\boldsymbol{x}^j$'s on the subspaces, with at least $K - 1$ points on each subspace.*

## 2.2. Estimation of the subspaces $\{S_i\}_{i=1}^n$

### 2.2.1. *GPCA as a polynomial factorization problem*

Proposition 1 and the linear system of equation (9) allow us to determine the number of subspaces $n$ and the coefficients $\boldsymbol{a}$ of the polynomial $p_n(\boldsymbol{x})$, respectively, from sample points $\{\boldsymbol{x}^j\}_{i=1}^N$. The rest of the problem becomes now how to recover $\{\boldsymbol{b}_i\}_{i=1}^n$ from $\boldsymbol{a}$.

From equations (3) and (7) we have that:

$$p_n(\boldsymbol{x}) = \sum a_{n_1,n_2,\ldots,n_K} x_1^{n_1} x_2^{n_2} \cdots x_K^{n_K} = \prod_{i=1}^n \left( \sum_{j=1}^K b_{ij} x_j \right).$$

Therefore, the problem of recovering $\{b_i\}_{i=1}^n$ from $a$ is equivalent to the following polynomial factorization problem:

PROBLEM 2 (Factorization of homogeneous polynomials). *Given a homogeneous polynomial $p_n(x) \in R_n(K)$, factorize it into $n$* distinct *polynomials in $R_1(K)$. We denote the space of polynomials which admit such a factorization as $R_n^F(K) \subset R_n(K)$, and call the elements of $R_n^F(K)$ factorable polynomials from now on.*

REMARK 2 (Factorization of symmetric tensors). *The polynomial factorization problem can also be interpreted as a tensor factorization problem: Given an $n$-th order symmetric tensor $\mathcal{V}$ in $Sym^n(\mathbb{R}^K)$, find vectors $v_1, v_2, \ldots, v_n \in \mathbb{R}^K$ such that*

$$\mathcal{V} = Sym(v_1 \otimes v_2 \otimes \ldots \otimes v_n) = \sum_{\sigma \in \mathfrak{S}_n} v_{\sigma(1)} \otimes v_{\sigma(2)} \otimes \ldots \otimes v_{\sigma(n)}$$

*Notice that*

$$\nu : \mathbb{R}^{K \times n} \rightarrow Sym^n(\mathbb{R}^K); \quad (v_1, v_2, \ldots, v_n) \mapsto Sym(v_1 \otimes v_2 \otimes \ldots \otimes v_n)$$

*maps a $K \times n$-dimensional space to a $M_n$-dimensional space. In general $M_n$ is much larger than $(K \times n - n + 1)$.*[3] *Therefore,* not *all symmetric tensors in the space $Sym^n(\mathbb{R}^K)$ can be factored in the above way.*

2.2.2. *Existence and uniqueness of the factorization*

Notice that an arbitrary element of $R_n(K)$ is *not* necessarily factorable into $n$ distinct elements of $R_1(K)$, e.g., the polynomial $x_1^2 + x_1x_2 + x_2^2$ is not. However, the existence of a factorization for $a$ is guaranteed by the definition of $p_n(x)$ as a product of linear functionals.

However, in practice the vector $a$ will be estimated from noise data, hence $p_n(x)$ will not necessarily be factorable. Therefore, we will need to address the issue of the existence of such a factorization and the projection of a non-factorable polynomial into a factorable one. We will delay our study of these two issues to later in this Section.

In relation to the uniqueness of the factorization, it is clear that each $b_i$ can be multiplied by an arbitrary scale to obtain the same $a$ up to scale. Since the scale of $a$ will be fixed to be 1 when solving (9), we are actually free to choose the scale of $n-1$ of the $b_i$'s only. The following proposition is a consequence of the well-known Gauss Lemma in Algebra (see page 181, [3]) and guarantees the uniqueness of the factorization of $p_n(x)$ up to $n-1$ scales:

[3] We here subtract $n-1$ parameters on the right is because we only have to consider unit vectors.

PROPOSITION 2 (Uniqueness of the factorization). *Since $\mathbb{R}$ is a factorial ring, the set of polynomials in $K$ variables $R[x_1, \ldots, x_K]$ is also factorial, that is any polynomial $p \in R[x_1, \ldots, x_K]$ has a unique factorization into irreducible elements. In particular, any element of $R_n(K) \subset R[x_1, \ldots, x_K]$ has a unique factorization.*

2.2.3. *Solving for the last 2 entries of each $\boldsymbol{b}_i$*

Knowing the existence and uniqueness of a solution to the polynomial factorization problem, we are now interested in finding an algorithm that recovers the $\boldsymbol{b}_i$'s from $\boldsymbol{a}$. To this end, we first consider the last $n+1$ coefficients of $p_n(\boldsymbol{x})$:

$$[a_{0,\ldots,0,n,0}\ ,\ a_{0,\ldots,0,n-1,1}\ ,\ \cdots\ ,\ a_{0,\ldots,0,0,n}]^T \in \mathbb{R}^{n+1}, \tag{12}$$

which define the following homogeneous polynomial of degree $n$ in the two variables $x_{K-1}$ and $x_K$:

$$\sum a_{0,\ldots,0,n_{K-1},n_K} x_{K-1}^{n_{K-1}} x_K^{n_K} = \prod_{i=1}^{n} \left(b_{iK-1} x_{K-1} + b_{iK} x_K\right). \tag{13}$$

Letting $y = x_{K-1}/x_K$, we have that:

$$\prod_{i=1}^{n} \left(b_{iK-1} x_{K-1} + b_{iK} x_K\right) = 0 \Leftrightarrow \prod_{i=1}^{n} \left(b_{iK-1} y + b_{iK}\right) = 0,$$

hence the $n$ roots of the polynomial

$$f_n(y) = a_{0,\ldots,0,n,0} y^n + a_{0,\ldots,0,n-1,1} y^{n-1} + \cdots + a_{0,\ldots,0,0,n} \tag{14}$$

are exactly $y_i = -b_{iK}/b_{iK-1}$, $i = 1, \ldots, n$. Therefore, after dividing $\boldsymbol{a}$ by $a_{0,\ldots,0,n,0}$, we obtain the last two entries of each $\boldsymbol{b}_i$ as:

$$(b_{iK-1}\ ,\ b_{iK}) = (1\ ,\ -y_i). \tag{15}$$

If $b_{iK-1} = 0$ for some $i$, then some of leading coefficients of $f_n(y)$ are zero and we cannot proceed as before, because $f_n(y)$ has less than $n$ roots. More specifically, assume that the first $\ell \leq n$ coefficients of $f_n(y)$ are zero and divide $\boldsymbol{a}$ by the $(\ell+1)$-st coefficient. In this case, we can choose $(b_{iK-1}, b_{iK}) = (0, 1)$, for $i = 1, \ldots, \ell$, and obtain $\{(b_{iK-1}, b_{iK})\}_{i=n-\ell+1}^{n}$ from the $n - \ell$ roots of $f_n(y)$ using equation (15). Finally, if all the coefficients of $f_n(y)$ are equal to zero, we set $(b_{iK-1}, b_{iK}) = (0, 0)$, for all $i = 1, \ldots, n$.

EXAMPLE 3. *Consider the case $n = 2$ and $K = 3$ illustrated in Example 2. The polynomial associated to the last $n+1$ entries of $\boldsymbol{a}$ is:*

$$b_{12}b_{23}y^2 + (b_{12}b_{23} + b_{13}b_{22})y + b_{13}b_{23} = (b_{12}y + b_{13})(b_{22}y + b_{23}) = 0, \tag{16}$$

*whose roots are:*

$$y_1 = -\frac{b_{13}}{b_{12}} \quad \text{and} \quad y_2 = -\frac{b_{23}}{b_{22}} \tag{17}$$

*from which we obtain both* $(b_{12}, b_{13})$ *and* $(b_{22}, b_{24})$ *up to scale.*

2.2.4. *Projection onto the space of factorable polynomials*

The previous section shows that the last two entries of each $\boldsymbol{b}_i$ can be computed up to scale from the coefficients of $p_n(\boldsymbol{x})$ associated to the last two variables. Furthermore, those coefficients define a polynomial of degree $n$ in one variable whose roots are always real. Notice that we could have chosen a polynomial associated to any other two variables and the roots of such a polynomial would also be real. Indeed, if one of the polynomials associated to any two variables has a complex root, $p_n(\boldsymbol{x})$ cannot be factorized as the product of polynomials of degree one. Unfortunately, this condition is only necessary but not sufficient for determining whether or not a general polynomial $p_n(\boldsymbol{x})$ in $R_n(K)$ can be factored into a product of polynomials of degree 1, as shown by the following example:

EXAMPLE 4 (Non-factorable polynomial with real roots). *Consider the polynomial* $p_2(\boldsymbol{x}) = x_1^2 + 4x_1x_2 + 4x_1x_3 + 2x_2^2 + 3x_2x_3 + x_3^2$ *in* $R_2(3)$. *Any sub-polynomial of* $p_2$ *in 2 variables has real roots. But* $p$ *is not factorizable into a product of two polynomials of degree 1.*

In practice, the polynomial $p_n(\boldsymbol{x})$ will be obtained from noisy data, and hence it may not be an element of $R_n^F(K)$. Therefore, we may consider the problem of finding a polynomial $p_n(\boldsymbol{x}) \in R_n^F(K)$ that is "close" to a given polynomial $q_n(\boldsymbol{x}) \in R_n(K)$. We notice that the space $R_n^F(K)$ is not convex, e.g., $2x_1^2 + x_1x_2 \in R_2^F(K)$ and $2x_2^2 - x_1x_2 \in R_2^F(K)$ but $x_1^2 + x_2^2 \notin R_2^F(K)$. We also notice that $R_n^F(K)$ can be described as an algebraic set in $\mathbb{R}^{M_n}$ as stated by the following proposition:

PROPOSITION 3 ($R_n^F(K)$ is a semi-algebraic set). *The set* $R_n^F(K)$ *is homeomorphic to the set*

$$\mathcal{F} = \{\boldsymbol{a} \in \mathbb{R}^{M_n} : h(\boldsymbol{a})\},$$

*where* $h$ *is a semi-algebraic formula.*

*Proof.* The coefficient vector $\boldsymbol{a} \in \mathbb{R}^{M_n}$ of a polynomial $p_n(\boldsymbol{a}, \boldsymbol{x})$ – here we explicitly write the dependency of $p_n$ on its coefficients $\boldsymbol{a}$ – which can be factored into a product of $n$ polynomials of degree 1 is described by the first-order formula

$$\exists \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{R}^K \quad \forall \boldsymbol{x} \in \mathbb{R}^K \quad p_n(\boldsymbol{a}, \boldsymbol{x}) = \prod_{i=1}^{n} (\boldsymbol{b}_i^T \boldsymbol{x}). \tag{18}$$

By Tarski-Seidendberg principle, after eliminating all the quantifiers, the set of all $\boldsymbol{a}$'s that have such a property must be semi-algebraic. That is, there is a semi-algebraic formula $h$ equivalent to (18) defining $\mathcal{F}$. ∎

Therefore, the problem of finding the factorable polynomial $p_n(\boldsymbol{x}) \in R_n^F(K)$ "closest to" $q_n(\boldsymbol{x}) \in R_n(K)$ is equivalent to:

$$\begin{array}{ll} \min & \|\boldsymbol{a} - \boldsymbol{c}\|_F^2 \\ \text{subject to} & \boldsymbol{a} \in \mathcal{F} \end{array} \tag{19}$$

where $\boldsymbol{a}$ and $\boldsymbol{c}$ are the coefficients of $p_n(\boldsymbol{x})$ and $q_n(\boldsymbol{x})$, respectively, and $\|\cdot\|_F$ is a norm in $\mathbb{R}^{M_n}$ defined as $\|\boldsymbol{a}\|_F^2 = \sum (n_1)! \cdots (n_K)! a^2_{n_1,\ldots,n_K}$. While it remains an open problem how to do the projection in general, there is a closed form solution for $n = 2$, as shown by the example below.

EXAMPLE 5 (Projection of a non-factorable polynomial). *Consider the polynomial* $q_2(\boldsymbol{x}) = c_{20}x_1^2 + c_{11}x_1x_2 + c_{02}x_2^2$ *and let* $\mathbf{c} = (c_{20}, c_{11}, c_{02})^T$. *In this case, the semi-albegraic formula defining* $R_n^F(K)$ *is given by* $h(\boldsymbol{a}) = a_{11}^2 - 4a_{20}a_{02} \geq 0$ *and the norm of* $\boldsymbol{a}$ *is given by* $\|\boldsymbol{a}\|_F^2 = 2a_{20}^2 + a_{11}^2 + 2c_{02}^2$. *The optimization problem can be solved using Lagrange multipliers to obtain the projection* $p_2(\boldsymbol{x}) = a_{20}x_1^2 + a_{11}x_1x_2 + a_{02}x_2^2$ *of* $q_2(\boldsymbol{x})$ *onto* $R_n^F(K)$, *with*

$$a_{20} = \frac{c_{20} + \lambda c_{02}}{1 - \lambda^2} \quad a_{11} = \frac{c_{11} - \lambda c_{11}}{1 - \lambda^2} \quad a_{02} = \frac{c_{02} + \lambda c_{20}}{1 - \lambda^2}, \tag{20}$$

*where*

$$\lambda = \frac{\min(c_{11}^2 - 4c_{20}c_{02}, 0)}{\|\boldsymbol{c}\|_F^2 + \sqrt{\|\boldsymbol{c}\|_F^4 - (c_2^2 - 4c_1c_2)^2}}.$$

2.2.5. *Solving for the first $K-2$ entries of each $\boldsymbol{b}_i$*

We have demonstrated how to obtain the last two entries of each $\boldsymbol{b}_i$ from the roots of a polynomial of degree $n$ in one variable. We are now left with the remaining $K-2$ entries of each $\boldsymbol{b}_i$. For the sake of simplicity, let us start with the following example:

EXAMPLE 6. *Consider the case $n = 2$ and $K = 3$ illustrated in Example 2. We know how to compute $b_{12}$, $b_{13}$, $b_{22}$ and $b_{24}$ as described in Example 3, and would like to compute $b_{11}$ and $b_{21}$. We notice that the coefficients of the monomials which are linear in $x_1$ ($a_{1,1,0}$ and $a_{1,0,1}$) are linear in $b_{11}$ and $b_{21}$. Thus we can write:*

$$\begin{bmatrix} b_{22} & b_{12} \\ b_{23} & b_{13} \end{bmatrix} \begin{bmatrix} b_{11} \\ b_{21} \end{bmatrix} = \begin{bmatrix} a_{1,1,0} \\ a_{1,0,1} \end{bmatrix} \tag{21}$$

*from which we can linearly solve for $b_{11}$ and $b_{21}$.*

We know generalize Example 6 to arbitrary $K$ and $n$. We assume that we have computed $b_{ij}$, $i = 1, \ldots, n$, $j = J+1, \ldots K$ for some $J$, starting with the case $J = K-2$, and show how to linearly solve for $b_{iJ}$, $i = 1, \ldots, n$. As in Example 6, the key is to consider the coefficients of $p_n(\boldsymbol{x})$ associated to monomials of the form $x_J\, x_{J+1}^{n_{J+1}} \cdots x_K^{n_K}$ which are *linear* in $x_J$. These coefficients are of the form $a_{0,\ldots,0,1,n_{J+1},\ldots,n_K}$ and are linear in $b_{iJ}$. Therefore, we can linearly solve for $b_{IJ}$ from:

$$\begin{bmatrix} \mathcal{V}_1^J & \mathcal{V}_2^J & \cdots & \mathcal{V}_n^J \end{bmatrix} \begin{bmatrix} b_{1J} \\ b_{2J} \\ \vdots \\ b_{nJ} \end{bmatrix} = \begin{bmatrix} a_{0,\ldots,0,1,n-1,0,\ldots,0} \\ a_{0,\ldots,0,1,n-2,1,\ldots,0} \\ \vdots \\ a_{0,\ldots,0,1,0,0,\ldots,n-1} \end{bmatrix} \tag{22}$$

where $\mathcal{V}_i^J$ are the coefficients of the following homogeneous polynomial of degree $n-1$ in $K-J$ variables:

$$g_i^J(\boldsymbol{x}) = \prod_{\ell=1}^{i-1} \left( \sum_{j=J+1}^{K} b_{\ell j} x_j \right) \prod_{\ell=i+1}^{n} \left( \sum_{j=J+1}^{K} b_{\ell j} x_j \right). \tag{23}$$

2.2.6. *Uniqueness of the solution of the factorization algorithm*
Equation (22) admits a solution by the definition of $\boldsymbol{a}$. Hence, the only question left is whether the solution is unique, or equivalently whether the vectors $\mathcal{V}_1^J, \ldots, \mathcal{V}_n^J$ are linearly independent. The following proposition gives a necessary and sufficient condition for uniqueness:

PROPOSITION 4 (Uniqueness of the solution given by the algorithm). *The vectors $\{\mathcal{V}_i^J\}_{i=1}^n$ are linearly independent if and only if for all $r \neq s$, $1 \leq r, s \leq n$, $(b_{rJ+1}, b_{rJ+2}, \ldots, b_{rK})$ and $(b_{sJ+1}, b_{sJ+2}, \ldots, b_{sK})$ are pairwise linearly independent. Furthermore, the vectors $\{\mathcal{V}_i^{K-2}\}_{i=1}^n$ are linearly independent if and only if the polynomial $f_n(y)$ has distinct roots and at most one of its leading coefficients is zero.*

*Proof.* We do the proof by induction on $n$. Let $h_i^J(\boldsymbol{x}) \triangleq \sum_{j=J+1}^K b_{ij} x_j$. By definition, the vectors $\mathcal{V}_i^J$ are linearly independent if

$$h^J \triangleq \sum_{i=1}^n \alpha_i h_1^J \cdots h_{i-1}^J h_{i+1}^J \cdots h_n^J = 0 \tag{24}$$

if and only if $\alpha_i = 0, \alpha_i \in \mathbb{R}$, $i = 1, \ldots, n$. If $n = 2$, (24) reduces to:

$$\alpha_1 h_2^J + \alpha_2 h_1^J = 0. \tag{25}$$

Therefore $\mathcal{V}_1^J$ is independent from $\mathcal{V}_2^J$ if and only if $h_1^J$ is independent from $h_2^J$, which happens if and only if $(b_{1J+1}, b_{1J+2}, \ldots, b_{1K})$ is independent from $(b_{2J+1}, b_{2J+2}, \ldots, b_{2K})$. Now assume the that the proposition is true for $n-1$. After dividing (24) by $h_1^J$ we obtain:

$$\frac{h^J}{h_1^J} = \alpha_1 \frac{h_2^J \cdots h_n^J}{h_1^J} + \sum_{i=2}^{n} \alpha_i \underbrace{h_2^J \cdots h_{i-1}^J h_{i+1}^J \cdots h_n^J}_{\text{polynomial in } R_{n-1}(K-J)} = 0. \tag{26}$$

If $\alpha_1 = 0$, then the proof reduces to the case $n-1$, which is true by the induction hypothesis. If $\alpha_1 \neq 0$, then $\frac{h_2^J \cdots h_n^J}{h_1^J}$ must belong to $R_{n-1}(K-J)$, which happens only if $h_1^J$ is proportional to some $h_i^J$, $i = 2, \ldots, n$, i.e. if $(b_{1J+1}, b_{1J+2}, \ldots, b_{1K})$ is proportional to some $(b_{iJ+1}, b_{iJ+2}, \ldots, b_{iK})$. The fact that the choice of $h_1^J$ as a divisor was arbitrary completes the proof of the first part. As for the second part, by construction the vectors $(b_{rK-1}, b_{rK})$ and $(b_{sK-1}, b_{sK})$ are independent if and only if the roots of $f_n(y)$ are distinct and $f_n(y)$ has at most one leading coefficient equal to zero. ∎

2.2.7. *Obtaning a unique solution for the degenerate cases*

Proposition 4 states that in order for the $K-2$ linear systems in (22) to have a unique solution, we must make sure that the polynomial $f_n(y)$ is non degenerate, i.e. $f_n(y)$ has no repeated roots and at most one of its leading coefficients is zero. One possible approach to avoid non-uniqueness is to choose a pair of variables $(x_j, x_{j'})$ for which the corresponding polynomial $f_n(y)$ is nondegenerate. The following proposition guarantees that we can do so if $n = 2$. Unfortunately the result is not true for $n > 2$ as shown by Example 7.

PROPOSITION 5 (Choosing a good pair of variables when $n = 2$). *Given the polynomial $p_2(\boldsymbol{x})$, there exist a pair of variables $(x_j, x_{j'})$ such that the associated polynomial $f_2(y)$ is nondegenerate.*

*Proof.* For the sake of contradiction, assume that for any pair of variables $(x_j, x_{j'})$ the associated polynomial $f_2(y)$ has a repeated root or the first two leading coefficients are zero. Proposition 4 implies that for all $j \neq j'$, $(b_{1j}, b_{1j'})$ is parallel to $(b_{2j}, b_{2j'})$, hence, all the $2 \times 2$ minors of the matrix $B = [\boldsymbol{b}_1\ \boldsymbol{b}_2]^T \in \mathbb{R}^{2 \times K}$ are equal to zero. This implies that $\boldsymbol{b}_1$ is parallel to $\boldsymbol{b}_2$, violating the assumption of distinct subspaces. ∎

EXAMPLE 7 (A polynomial with repeated roots). *Consider the following polynomial in $R_3(3)$:*

$$p_3(\boldsymbol{x}) = (x_1 + x_2 + x_3)(x_1 + 2x_2 + 2x_3)(x_1 + 2x_2 + x_3).$$

*The associated polynomials in two variables are* $4x_2^3+10x_2^2x_3+8x_2x_3^2+2x_3^2$, $x_1^3+4x_1^2x_3+5x_1x_3^2+2x_3^2$ *and* $x_1^3+5x_1^2x_2+8x_1x_2^2+4x_2^2$, *all of them having repeated roots.*

We conclude that, even though the uniqueness of the factorization is guaranteed by Proposition 2, there are some cases for $n > 2$ for which our factorization algorithm (based on solving for the roots a polynomial of degree $n$ in one variable plus $K-2$ linear systems in $n$ variables) will not be able to provide the *unique* solution. The reason for this is that our algorithm is not using *all* the coefficients in $\boldsymbol{a}$, but only the ones for which the problem is linear.

One possible algorithm to obtain a unique solution for these degenerate cases is to solve polynomials of degree $r$ in $n-r$ variables. We will not pursue this direction here. Instead, we will try to find a linear transformation on $\boldsymbol{x}$, hence on the $b_{ij}$'s, that gives a new vector of coefficients $\boldsymbol{a}'$ whose associated polynomial $f_n'(y)$ is non-degenerate. It is clear that we only need to modify the entries of each $\boldsymbol{b}_i$ associated to the last two variables. Thus, we consider the following linear transformation $L : \mathbb{R}^K \to \mathbb{R}^K$:

$$\boldsymbol{x} = L\boldsymbol{y} = \begin{bmatrix} 1 & 0 & \cdots & 0 & t & t \\ 0 & 1 & & 0 & t & t \\ \vdots & & \ddots & & & \vdots \\ & & & 1 & t & t \\ & & & & 1 & t \\ 0 & 0 & \cdots & & 0 & 1 \end{bmatrix} \boldsymbol{y}. \tag{27}$$

Under this transformation, the polynomial $p_n(\boldsymbol{x})$ becomes:

$$p_n'(\boldsymbol{y}) = p_n(L\boldsymbol{y}) = \prod_{i=1}^{n} \boldsymbol{b}_i^T L\boldsymbol{y} =$$

$$\prod_{i=1}^{n}\left(\sum_{j=1}^{K-1} b_{ij}y_j + \underbrace{\left[t\sum_{j=1}^{K-2} b_{ij} + b_{iK-1}\right]}_{b'_{iK-1}(t)} y_{K-1} + \underbrace{\left[t\sum_{j=1}^{K-1} b_{ij} + b_{iK}\right]}_{b'_{iK}(t)} y_K\right)$$

Therefore, the polynomial associated to $y_{K-1}$ and $y_K$ will have distinct roots for all $t \in \mathbb{R}$, except for the $t$'s which are roots of the following second order polynomial:

$$b'_{rK-1}(t)b'_{sK}(t) = b'_{sK-1}(t)b'_{rK}(t) \tag{28}$$

for some $r \neq s$, $1 \leq r, s \leq n$. Since there are a total of $n(n+1)/2$ such polynomials, each of them having at most 2 roots, we can choose $t$ arbitrarily, except for $n(n+1)$ values.

Once $t$ has been chosen, we need to compute the coefficients $\boldsymbol{a}'$ of the new polynomial $p_n'$. The following proposition establishes the relationship between $\boldsymbol{a}$ and $\boldsymbol{a}'$:

PROPOSITION 6. *Let $\boldsymbol{a}$ and $\boldsymbol{a}'$ be the coefficients of the polynomials $p_n(\boldsymbol{x}) \in R_n(K)$ and $p_n'(\boldsymbol{y}) = p_n(L\boldsymbol{x}) \in R_n(K)$, respectively, where $L : \mathbb{R}^K \to \mathbb{R}^K$ is a non-singular linear map. Then $L$ induces a linear transformation $T : \mathbb{R}^{M_n} \to \mathbb{R}^{M_n}$, $\boldsymbol{a} \mapsto \boldsymbol{a}' = T\boldsymbol{a}$. Furthermore, the column of $T$ associated to $a_{n_1,n_2,\ldots,n_K}$ is given by the coefficients of the polynomial:*

$$(\boldsymbol{\ell}_1^T \boldsymbol{y})^{n_1} \, (\boldsymbol{\ell}_2^T \boldsymbol{y})^{n_2} \, \cdots \, (\boldsymbol{\ell}_K^T \boldsymbol{y})^{n_K},$$

*where $\boldsymbol{\ell}_j^T$ is the $j$-th row of $L$.*

*Proof.* Let $p_n(\boldsymbol{x}), q_n(\boldsymbol{x}) \in R_n(K)$ and $\alpha, \beta \in \mathbb{R}$. Then the polynomial $\alpha p_n(\boldsymbol{x}) + \beta q_n(\boldsymbol{x})$ is transformed by $L$ into $\alpha p_n(L\boldsymbol{y}) + \beta q_n(L\boldsymbol{y})$. Therefore $T$ is linear. Now in order to find the column of $T$ associated to $a_{n_1,n_2,\ldots,n_K}$, we just need to apply the transformation $T$ to the monomial $x_1^{n_1} x_2^{n_2} \cdots x_K^{n_K} = (e_1^T \boldsymbol{x})^{n_1} (e_2^T \boldsymbol{x})^{n_2} \cdots (e_K^T \boldsymbol{x})^{n_K}$, where $\{e_j\}_{j=1}^K$ is the standard basis for $\mathbb{R}^K$. We obtain $(e_1^T L\boldsymbol{y})^{n_1} (e_2^T L\boldsymbol{y})^{n_2} \cdots (e_K^T L\boldsymbol{y})^{n_K}$, that is, $(\boldsymbol{\ell}_1^T \boldsymbol{y})^{n_1} \, (\boldsymbol{\ell}_2^T \boldsymbol{y})^{n_2} \, \cdots \, (\boldsymbol{\ell}_K^T \boldsymbol{y})^{n_K}$. ■

REMARK 3. *Due to the upper triangular structure of $L$ in (27), the matrix $T$ will be lower triangular. Furthermore, since each entry of $L$ is a polynomial of degree at most 1 in $t$, the entries of $T$ will be polynomials of degree at most $n$ in $t$.*

By construction, the polynomial $f_n'(y)$ associated to the last two variables of $p_n'(\boldsymbol{y})$ will have no repeated roots. Threrefore, we can apply the previously described factorization algorithm to the coefficients $\boldsymbol{a}'$ of $p_n'(\boldsymbol{y})$ to obtain the set of transformed normal vectors $\{\boldsymbol{b}_i'\}_{i=1}^n$. Since by definition of $p_n'$ we have $\boldsymbol{b}_i'^T = \boldsymbol{b}_i^T L$, the original normal vectors are given by $\boldsymbol{b}_i = L^{-T} \boldsymbol{b}_i'$. It turns out that, due to the particular structure of $L$, we do not actually need to compute $L^{-T}$. We can obtain $\{\boldsymbol{b}_i\}_{i=1}^n$ directly from $\{\boldsymbol{b}_i'\}_{i=1}^n$ and $t$ as follows:

$$\begin{array}{lll} b_{ij} = b_{ij}', & i = 1, \ldots, n, \; j = 1, \ldots, K-2 \\ b_{iK-1} = b_{iK-1}' - t \sum_{j=1}^{K-2} b_{ij}, & i = 1, \ldots, n \\ b_{iK} = b_{iK}' - t \sum_{j=1}^{K-1} b_{ij}, & i = 1, \ldots, n. \end{array} \tag{29}$$

EXAMPLE 8. *Let $n = 3$ and $K = 3$. Then $L$ and $T$ are given by:*

$$L = \begin{bmatrix} 1 & t & t \\ 0 & 1 & t \\ 0 & 0 & 1 \end{bmatrix} \qquad \textit{and} \tag{30}$$

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3t & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3t & t & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3t^2 & 2t & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 6t^2 & 2t^2+2t & 2t & 2t & 1 & 0 & 0 & 0 & 0 & 0 \\ 3t^2 & 2t^2 & 2t & t^2 & t & 1 & 0 & 0 & 0 & 0 \\ t^3 & t^2 & 0 & t & 0 & 0 & 1 & 0 & 0 & 0 \\ 3t^3 & t^3+2t^2 & t^2 & 2t^2+t & t & 0 & 3t & 1 & 0 & 0 \\ 3t^3 & 2t^3+t^2 & 2t^2 & t^3+2t^2 & t^2+t & t & 3t^2 & 2t & 1 & 0 \\ t^3 & t^3 & t^2 & t^3 & t^2 & t & t^3 & t^2 & t & 1 \end{bmatrix}. \tag{31}$$

*Notice that if $p_3(\boldsymbol{x}) = a_6x_1x_3^2 + a_9x_2x_3^2 + a_{10}x_3^3$, then the transformed polynomial is $p_n'(y) = a_6y_1y_3 + (a_9 + ta_6)y_2y_3^2 + (a_10 + ta_6 + ta_9)y_3^3$. Thus, the polynomial associated to the last two variables is $f_n'(y) = (a_9 + ta_6)y + a_{10} + t(a_6 + a_9)$, which has more than one leading coefficient equal to zero for all $t$. This is not a problem, because the given polynomial, $p_3(\boldsymbol{x})$, is non-factorable.*

### 2.2.8. *GPCA algorithm*

We summarize the results of this section with the following GPCA algorithm

ALGORITHM 1 (GPCA algorithm). *Given sample points $\{\boldsymbol{x}^j\}_{j=1}^N$, find the number of subspaces $n$ and the normal to the subspaces $\{\boldsymbol{b}_i\}_{i=1}^n$ as follows:*

1. *Apply the Veronese map of order $i$, for $i = 1, 2, \ldots$, to the vectors $\{\boldsymbol{x}^j\}_{j=1}^N$ and form the matrix $L_i$ in (9). Stop when rank($L_i$) $= M_i - 1$ and set the number of subspaces $n$ to be the current $i$. Then solve for $\boldsymbol{a}$ from $L_n\boldsymbol{a} = 0$ and normalize so that $\|\boldsymbol{a}\| = 1$.*

2. *a) Divide the resulting $\boldsymbol{a}$ by the first nonzero coefficient of $f_n(y)$.*

   *b) If the first $\ell$, $0 \le \ell \le n$, coefficients of $f_n(y)$ are equal to zero, set $(b_{iK-1}, b_{iK}) = (0,1)$ for $i = 1, \ldots, \ell$. Then use (15) to compute $\{(b_{iK-1}, b_{iK})\}_{i=n-\ell+1}^n$ from the $n - \ell$ roots of $f_n(y)$.*

   *c) If all the coefficients of $f_n(y)$ are zero, set $(b_{iK-1}, b_{iK}) = (0,1)$, for $i = 1, \ldots, n$.*

*d) If $(b_{rK-1}, b_{rK})$ is parallel to $(b_{sK-1}, b_{sK})$ for some $r \neq s$, apply the transformation $x = Ly$ in (27) and repeat 2a), 2b) and 2c) for the transformed polynomial $p'_n(y)$ to obtain $\{(b'_{iK-1}, b'_{iK})\}_{i=1}^n$.*

*3. Given $(b_{iK-1}, b_{iK})$, $i = 1, \ldots, n$, solve for $\{b_{iJ}\}_{i=1}^n$ from (22) for $J = K-2, \ldots, 1$. If a transformation $L$ was used in 2 d), then compute $b_i$ from $b'_i$ and $t$ using equation (29).*

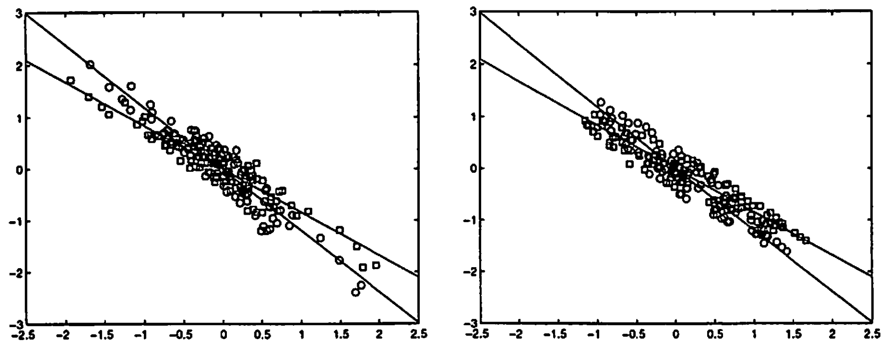
## 3. Estimation of a Mixture of Probabilistic Models

In this section, we apply the theory of GPCA to the estimation of subspaces for a mixture of probabilistic models. We assume that the sample points $\{x^j\}_{j=1}^N$ are drawn from a certain distribution (Gaussian or uniform) in $\mathbb{R}^K$ that is approximately degenerate, i.e. there is at least one direction in which the variance of the data is approximately equal to zero. The extreme case in which the variance is equal to zero in a certain direction(s) corresponds to the ideal case for GPCA in which the data lies exactly in a linear subspace of $\mathbb{R}^K$ which is orthogonal to the zero variance directions.

In our experiments, we will consider mixtures of $n = 2$ and $n = 4$ probabilistic models in a $K = 2$ dimensional space. In all cases the mean is assumed to be zero, and the standard deviation is 1 along the direction of the subspace and $e < 1$ along the direction orthogonal to the subspace, where the parameter $e$ represents the eccentricity of the covariance matrix[4]. In all cases, we will choose the sample points in subspace $n > 1$ to be a rotated version of those in subspace 1. Therefore, we will use data that is purposely *statistically dependent.*

Figures 2a) and 2b) show mixtures of $n = 2$ Gaussian and Uniform distributions, respectively, with the main direction of each subspace superimposed. The eccentricity of the covariance matrix is $e = 0.15$ and the angle between the two subspaces of $\theta = 10°$. We observe that it is quite difficult to group data points belonging to the same subspace, even if we knew the true direction for each subspace. This is particularly true for the Gaussian distribution, because it is zero mean, and hence most of the data for both subspaces is concentrated around the origin. Therefore, we should not expect GPCA to give good segmentation results for this type of data. Instead, we should expect GPCA to give a good estimate of the direction of the subspaces in spite of the grouping, because GPCA does not use the grouping to estimate the subspaces as most algorithms do.

[4] The eccentricity is defined as the square root of the ratio between the smallest and the largest singular value of the covariance matrix $\Sigma$.




(a) Gaussian distribution (b) Uniform distribution

*Figure 2.* A mixture of $n = 2$ probabilistic models with a) Gaussian and b) Uniform distributions. The number of data points is 100 per subspace. Points in susbpace 1 are denoted with a $\circ$ and points in subspace 2 are denoted with a $\square$. The eccentricity of the covariance matrix for both distributions is $e = 0.15$ and the angle between the two subspaces of $\theta = 10°$.

Figure 3 presents simulation results for a mixture of 2 Gaussian distributions with an eccentricity of 0.15 for different values of the angles between the two subspaces $\theta$. We observe that there is a high rate of missclassification (around 20 to 35%) for small $\theta$. As $\theta$ increases, the missclasification reduces to 10% approximately. In spite of these high missclasification rates, the estimation of a basis for the subspaces is remarkably good: the error in the estimation of the subspaces is approximately of 1°, with a maximum error of 2.1° for an angle between subspaces of only $\theta = 10°$.

Figure 4 presents simulation results for a mixture of 2 Gaussian distributions with an angle between subspaces of $45^{\text{deg}}$ for different values of the the eccentricity. As expected, the rate of missclassification increases with the exentricity of the covariance matrix and so does the error in the estimation of the subspaces. Again, in spite of extremely high missclasification rates, the estimation of a basis for the subspaces is obtained with an error of less than 5°.

## 4. Two-View Multibody Structure from Motion

In this section, we apply GPCA to the problem of estimating the 3D motion of multiple moving objects from 2 perspective views. We here give a brief summary on how GPCA can be used to solve the problem and refer the reader to [6] for the details.

Assume we are given with a pair of images $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ of a point $p$ undergoing one out of possibly many motions, say $\{(R_i, T_i)\}_{i=1}^n$, with

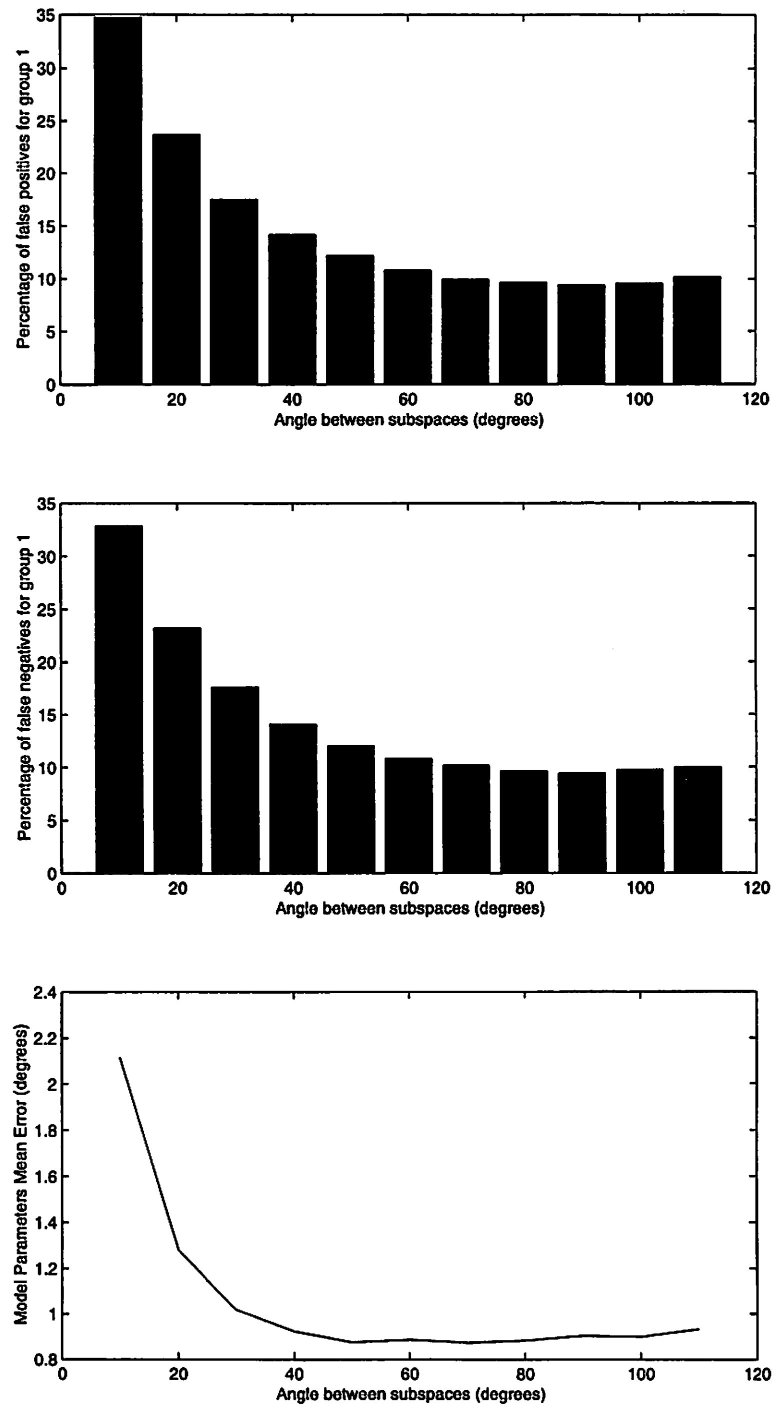



*Figure 3.* Estimation of the subspaces for a mixture of two Gaussians as a function of the angle between the subspaces. The number of data points is 100 for each Gaussian, and the eccentricity of the covariance matrix is 0.15.

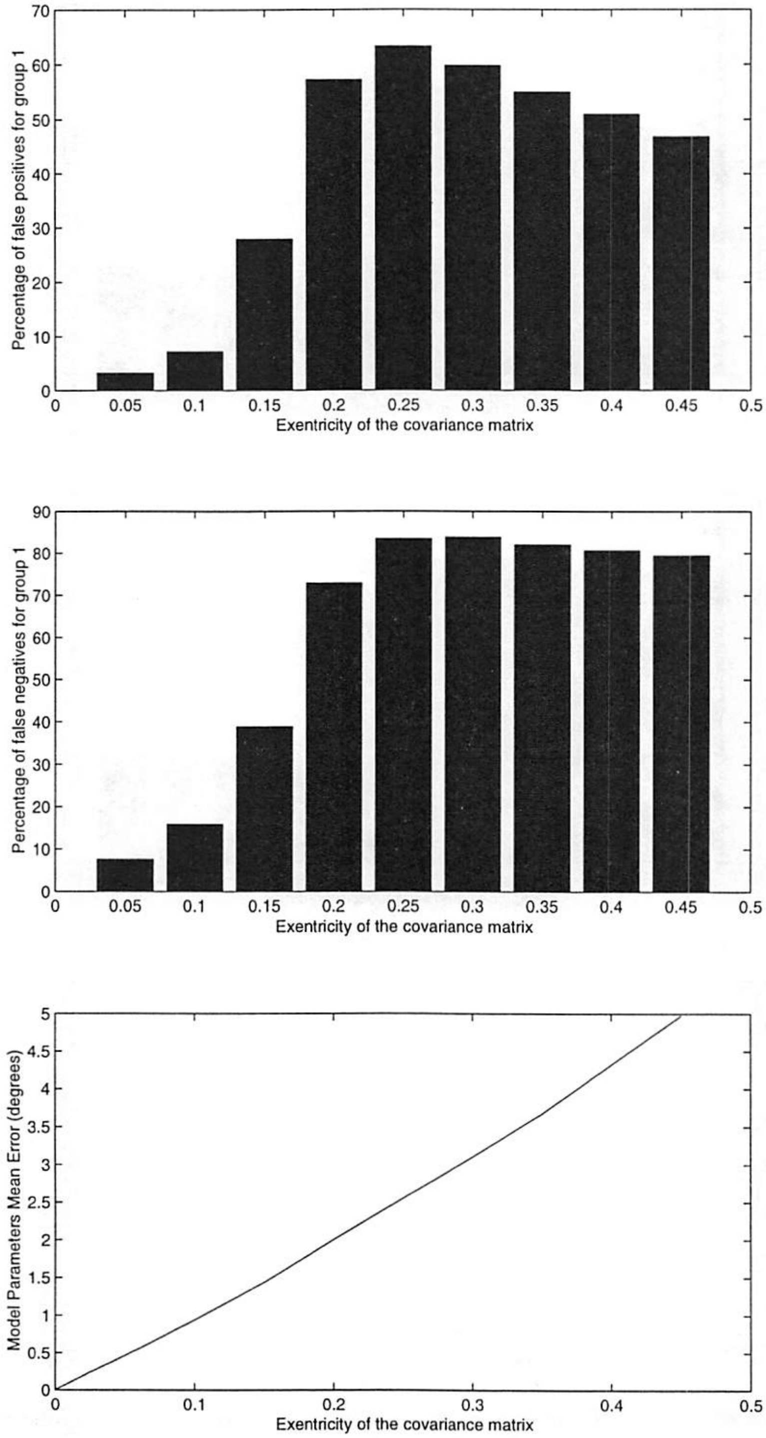



*Figure 4.* Estimation of the subspaces for a mixture of four Gaussians as a function of the eccentricity of the covariance matrix. The number of data points is 100 for each Gaussian, and the angle between subspaces is 45°.

$R_i \in \mathbb{R}^{3\times 3}$ and $T_i \neq 0 \in \mathbb{R}^3$ representing the camera rotation and translation (calibrated case) or the homography and epipole (uncalibrated), respectively. If $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is undergoing motion $i$, then it must satisfy the so-called epipolar constraint $\boldsymbol{x}_2^T F_i \boldsymbol{x}_1 = 0$, where $F_i = \widehat{T}_i R_i \in \mathbb{R}^{3\times 3}$ is the fundamental matrix associated to motion $i$. Since we do not know the motion associated to a given an image pair $(\boldsymbol{x}_1, \boldsymbol{x}_2)$, as we did in (3), we can only enforce the following constraint:

$$\boxed{\prod_{i=1}^{n} \left(\boldsymbol{x}_2^T F_i \boldsymbol{x}_1\right) = 0,} \tag{32}$$

which we call the *multibody epipolar constraint*

While the GPCA constraint (3) is a product of linear forms, the multibody epipolar constraint is a product of *bilinear* forms, which makes the multibody structure from motion a much harder problem. It is true that one can convert each epipolar constraint $\boldsymbol{x}_2^T F_i \boldsymbol{x}_1 = 0$ into the linear constraint $(\boldsymbol{x}_2 \otimes \boldsymbol{x}_1)^T \boldsymbol{f}_i$, where $\boldsymbol{f}_i \in \mathbb{R}^9$ is the stack of the columns of $F_i$. However, it is not possible to apply GPCA with $K = 9$ to the resulting product of linear forms, because $\nu_n(\mathbb{R}^3 \otimes \mathbb{R}^3) \neq \nu_n(\mathbb{R}^9)$ as shown in [6]. Instead, we use a clever geometric analysis of the multibody structure from motion problem to convert the bilinear factorization problem into two GPCA problems with $K = 3$, as described below:

1. *Multibody fundamental matrix*: The Veronese map $\nu_n$ can be used to convert the multibody epipolar constraint (32) into the following bilinear constraint in $\nu_n(\boldsymbol{x}_1)$ and $\nu_n(\boldsymbol{x}_2)$:

$$\boxed{\nu_n(\boldsymbol{x}_2)^T \mathcal{F} \nu_n(\boldsymbol{x}_1) = 0,} \tag{33}$$

   where $\mathcal{F}$, the so-called *multibody fundamental matrix*, is a matrix in $\mathbb{R}^{M_n \times M_n}$ with $M_n = (n+1)(n+2)/2$. Each entry of $\mathcal{F}$ is a symmetric multilinear function of the entries of $F_i$. Notice that equation (33) can be re-written in linear form as $(\nu_n(\boldsymbol{x}_2) \otimes \nu_n(\boldsymbol{x}_1))^T \boldsymbol{f} = 0$, where $\boldsymbol{f}$ is the stack of the columns of $\mathcal{F}$.

2. *Estimation of $n$ and $\mathcal{F}$*: Given a set of $N$ image pairs $\{(\boldsymbol{x}_1^j, \boldsymbol{x}_2^j)\}_{j=1}^N$, one can form a matrix $L_i \in \mathbb{R}^{N \times M_n^2}$ whose $j$-th row is given by $(\nu_i(\boldsymbol{x}_2^j) \otimes \nu_i(\boldsymbol{x}_1^j))^T$. From this matrix, one can determine the number of motions $n$ as $\min\{i : \text{rank}(L_i) = N_i^2 - 1\}$, and the multibody fundamental matrix $\mathcal{F}$ from the linear system $L_n \boldsymbol{f} = 0$.

3. *Estimation of epipolar lines*: Given an image $\boldsymbol{x}_1$ in the first frame, the epipolar lines associated to it are defined as $\boldsymbol{\ell}_i = F_i \boldsymbol{x}_1$ and

one of such lines passes through the corresponding image $x_2$ in the second frame, i.e. $\ell_i^T x_2 = 0$. Now given the multibody fundamental matrix $F$, one can prove that $F\nu_n(x_1) \in \mathbb{R}^{M_n}$ represents the coefficients of the following homogeneous polynomial in $x$: $f(x) = (\ell_1^T x)\cdots(\ell_n^T x)$. Therefore, one can compute the epipolar lines $\{\ell_i\}_{i=1}^n$ associated to any image point $x_1$ using steps 2 and 3 of the GPCA Algorithm 1 for $K = 3$.

4. *Estimation of epipoles*: The epipoles are the vectors $\{T_i\}_{i=1}^n$ such that $T_i^T F_i = 0$. Thus, given an arbitrary epipolar line $\ell$, there exist one epipole $T_i$ such that $T_i^T \ell = 0$, which implies that $\ell$ satisfies the following polynomial: $g(\ell) = (T_1^T \ell)\cdots(T_n^T \ell) = 0$. Therefore, given a set of epipolar lines $\{\ell^j\}_{j=1}^m$ computed from the previous step, one can apply to them the GPCA Algorithm 1 with $K = n$ to obtain the epipoles $\{T_i\}_{i=1}^n$, provided that the epipoles are distinct.

5. *Estimation of fundamental matrices*: Let $f_i^1 \in \mathbb{R}^3$, $f_i^2 \in \mathbb{R}^3$ and $f_i^3 \in \mathbb{R}^3$ be the columns of the fundamental matrix $F_i \in \mathbb{R}^{3\times 3}$ associated to motion $i$. Given $x = (x_1, x_2, x_3) \in \mathbb{R}^3$, one can prove that the vector $F\nu_n(x) \in \mathbb{R}^{M_n}$ represents the coefficients of the following homogeneous polynomial in $y$:

$$h(y) = \left((x_1 f_1^1 + x_2 f_1^2 + x_3 f_1^3)^T y\right)\cdots\left((x_1 f_n^1 + x_2 f_n^2 + x_3 f_n^3)^T y\right).$$

Therefore, given $F$ one can estimate the columns of each fundamental matrix and more generally any linear combination of them $\{x_1 f_i^1 + x_2 f_i^2 + x_3 f_i^3\}_{i=1}^n$ (up to scale) using steps 2 and 3 of the GPCA Algorithm 1 with $K = 3$. However:

a) We do not know the fundamental matrix to which the recovered vectors belong to, because the GPCA problem is symmetric;

b) The recovered vectors are obtained up to scale only.

The first problem is easily solvable: if the recovered vector corresponds to $F_i$, then it must be perpendicular to the previously computed $T_i$ (again we assume that the epipoles are distinct). As for the second problem, for each $i$ let $\ell_i^j$, $j = 1, \ldots, m$ be the recovered vector corresponding to $x^j$ that is perpendicular to $T_i$. The $x^j$'s can be chosen arbitraryly, but we choose $x^1 = (1,0,0)^T$, $x^2 = (0,1,0)$ and $x^3 = (0,0,1)^T$. Then there exist unknown scales $\lambda_i^j$ such that:

$$\begin{aligned} \lambda_i^j \ell_i^j &= x_1^j f_i^1 \quad + x_2^j f_i^2 \quad + x_3^j f_i^3, & j \geq 4 \\ &= x_1^j(\lambda_i^1 \ell_i^1) + x_2^j(\lambda_i^2 \ell_i^2) + x_3^j(\lambda_i^3 \ell_i^3), & j \geq 4. \end{aligned}$$

The fundamental matrices are then given by:

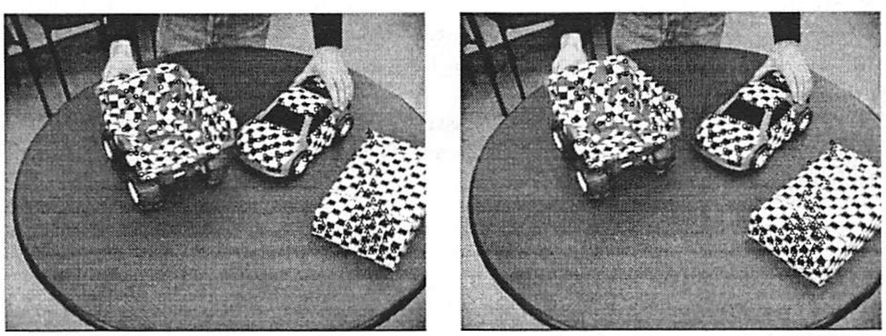
$$F_i = [\boldsymbol{f}_i^1 \; \boldsymbol{f}_i^2 \; \boldsymbol{f}_i^3] = [\lambda_i^1 \boldsymbol{\ell}_i^1 \; \lambda_i^2 \boldsymbol{\ell}_i^2 \; \lambda_i^3 \boldsymbol{\ell}_i^3], \tag{34}$$

where $\lambda_i^1$, $\lambda_i^2$ and $\lambda_i^3$ are obtained as the solution of the linear system:

$$\begin{bmatrix} \widehat{\boldsymbol{\ell}_i^4}[x_1^4 \boldsymbol{\ell}_i^1 \; x_2^4 \boldsymbol{f}_i^2 \; x_3^4 \boldsymbol{\ell}_i^3] \\ \widehat{\boldsymbol{\ell}_i^4}[x_1^5 \boldsymbol{\ell}_i^1 \; x_2^5 \boldsymbol{f}_i^2 \; x_3^5 \boldsymbol{\ell}_i^3] \\ \vdots \\ \widehat{\boldsymbol{\ell}_i^4}[x_1^m \boldsymbol{\ell}_i^1 \; x_2^m \boldsymbol{f}_i^2 \; x_3^m \boldsymbol{\ell}_i^3] \end{bmatrix} \begin{bmatrix} \lambda_i^1 \\ \lambda_i^2 \\ \lambda_i^3 \end{bmatrix} = 0 \tag{35}$$

6. *Motion Segmentation*: Given the fundamental matrices, the image pair $(\boldsymbol{x}_1^j, \boldsymbol{x}_2^j)^j$ is assigned to group $i$ if $\boldsymbol{x}_2^j F_i \boldsymbol{x}_1^j = 0$. In the presence of noise, the image pair is assigned to the group $i$ that minimizes $(\boldsymbol{x}_2^j F_i \boldsymbol{x}_1^j)^2$.

We evaluated the proposed approach to segment a real image sequence with $n = 3$ moving objects: a truck, a car and a box. Figure 4 shows two frames of the sequence with the tracked features superimposed. We used the algorithm in [1] to track a total of $N = 173$ point features: 44 for the truck, 48 for the car and 81 for the box. Figure 4 plots the segmentation of the image points obtained through GPCA. The obtained segmentation has no mismatches.



(a) First frame (b) Second frame

*Figure 5.* A motion sequence with two cars and a box. Tracked features are marked with a 'o' for the first car, a '□' for the second car and an '△' for the box.

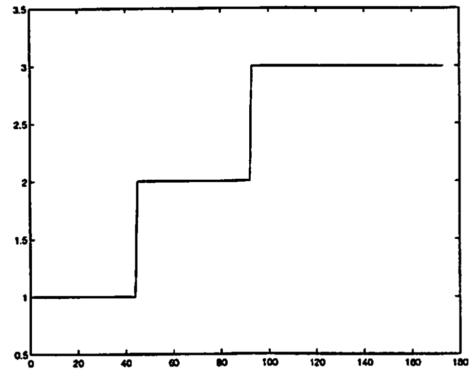



*Figure 6.* Motion segmentation results. Each image pair is assigned to the fundamental matrix for which the algebraic error is minimized. The first 44 points correspond to the first car, the next 48 to the second, and the last 81 to the box.

## References

1. A. Chiuso, P. Favaro, H. Jin, and S. Soatto. Motion and structure causally integrated over time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):523–535, April 2002.
2. L. de Lathauwer, B. de Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM Journal of Matrix Analysis*, 21(4):1253–1278, 2000.
3. S. Lang. *Algebra*. Addison-Wesley, third edition, 1993.
4. S. Soatto, G. Doretto, and Y. Wu. Dynamic textures. In *IEEE International Conference on Computer Vision*, pages 439–446, 2001.
5. M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.
6. R. Vidal, Y. Ma, S. Soatto, and S. Sastry. Multibody structure from motion from two perspective views. *International Journal of Computer Vision*, 2002. Submitted.