# An Unsupervised Learning Approach for Facial Expression Recognition using Semi-Definite Programming and Generalized Principal Component Analysis

Behnood Gholami,[a] Wassim M. Haddad,[a] and Allen R. Tannenbaum[b]

[a] School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, 30332
[b] Schools of Electrical & Computer and Biomedical Engineering, Georgia Institute of Technology, Atlanta, GA, 30332

## ABSTRACT

In this paper, we consider facial expression recognition using an unsupervised learning framework. Specifically, given a data set composed of a number of facial images of the same subject with different facial expressions, the algorithm segments the data set into groups corresponding to different facial expressions. Each facial image can be regarded as a point in a high-dimensional space, and the collection of images of the same subject resides on a manifold within this space. We show that different facial expressions reside on distinct subspaces if the manifold is unfolded. In particular, semi-definite embedding is used to reduce the dimensionality and unfold the manifold of facial images. Next, generalized principal component analysis is used to fit a series of subspaces to the data points and associate each data point to a subspace. Data points that belong to the same subspace are shown to belong to the same facial expression.

**Keywords:** Facial expression recognition, unsupervised learning, dimension reduction, semi-definite programming, manifold unfolding, principal component analysis

## 1. INTRODUCTION

The human face is a rich medium through which people communicate their emotions. Researchers have identified six basic human expressions, namely, happiness, sadness, anger, disgust, fear, and surprise.[1] Automatic facial expression recognition algorithms can be used in systems involving human-computer interaction.[2] An emerging field of application for facial expression recognition algorithms involves clinical decision support systems.[3,4] Specifically, the authors in Refs. 5 and 6 present a framework for assessing pain and pain intensity in neonates using digital imaging.

Among different approaches proposed for facial expression recognition are *manifold-based methods*.[7] In these methods, the facial image can be regarded as a point in a $D$-dimensional space (which is referred to as the *ambient space*), where $D$ is the number of pixels in the image or the number of parameters in a face model. The underlying assumption in manifold-based methods is that a set of facial images of a subject, which are represented by a set of points in a high-dimensional ambient space, resides on an intrinsically low-dimensional manifold. Hence, an important part of the facial expression recognition algorithm in such methods involve finding the manifold of facial expressions.

In this paper, we propose an unsupervised learning approach to facial expression recognition, where we show that different facial expressions reside on distinct subspaces if the manifold of facial images is unfolded.[8] Specifically, we introduce a new manifold-based method, where we use a maximum variance unfolding (MVU) approach[8] to identify the low-dimensional manifold of facial images and unfold it. Next, generalized principal component analysis is used to fit a series of subspaces to the data points and associate each data point to a subspace. Data points that belong to the same subspace are shown to belong to the same facial expression.

Further author information: (Send correspondence to B.G.)
B.G.: E-mail: behnood@gatech.edu, Telephone: (404) 894-3474
W.M.H: E-mail: wm.haddad@aerospace.gatech.edu, Telephone: (404) 894-1078
A.R.T.: E-mail: tannenba@ece.gatech.edu, Telephone: (404) 894-7582

Image Processing: Algorithms and Systems VIII, edited by Jaakko T. Astola, Karen O. Egiazarian,
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 7532, 75320K · © 2010 SPIE-IS&T
CCC code: 0277-786X/10/$18 · doi: 10.1117/12.839982

The contents of the paper are as follows. First, we review the MVU dimension reduction technique, which involves semi-definite programming and convex optimization. In Section 3, we review the generalized principal component analysis (GPCA). This framework uses algebro-geometric concepts to address the problem of data segmentation and subspace identification for a given set of data points. In Section 4, the MVU and GPCA methods are used to recognize facial expressions from a given set of images within an unsupervised learning framework. Finally, we draw conclusions in Section 5.

The notation used in this paper is fairly standard. Specifically, $\mathbb{Z}_+$ denotes the set of positive integers, $\mathbb{R}$ denotes the set of real numbers, $(\cdot)^{\mathrm{T}}$ denotes transpose, and $(\cdot)^\dagger$ denotes the Moore-Penrose generalized inverse. Furthermore, we write $\mathrm{tr}(\cdot)$ for the trace operator, $\mathcal{N}(\cdot)$ for null space, $\|\cdot\|$ for the Euclidean norm, and $\dim(\mathcal{S})$ for the dimension of a set $\mathcal{S} \subset \mathbb{R}^D$, where $D \in \mathbb{Z}_+$.

## 2. MANIFOLD UNFOLDING AND DIMENSION REDUCTION

In this section, we introduce the method of maximum variance unfolding (MVU), which involves a dimension reduction technique that uses semi-definite programming and convex optimization. Given a set of points sampled from a low-dimensional manifold in a high-dimensional ambient space, this technique unfolds the manifold (and hence, the points it contains) while preserving the local geometrical properties of the manifold.[8] This method can be regarded as a nonlinear generalization of the principal component analysis (PCA).[8]

Given a set of points in a high-dimensional ambient space, principal component analysis identifies a low-dimensional subspace such that the variance of the projection of the points on this subspace is maximized. More specifically, the basis of the subspace on which the projection of the points has the maximum variance is the eigenvectors corresponding to the non-zero eigenvalues of the covariance matrix.[9] In the case where the data is noisy, the singular vectors corresponding to the dominant singular values of the covariance matrix are selected.[10,11]

Given the set of $N$ input points $\mathcal{X} = \{x_1, x_2, \ldots, x_N\} \subset \mathbb{R}^D$, where $D$ is the dimension of the ambient space, we seek to find the set of $N$ output points $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^d$ such that $d < D$, $\mathcal{X}$ and $\mathcal{Y}$ are equivalent, and points sufficiently close to each other in the input data set $\mathcal{X}$ remain sufficiently close in the output data set $\mathcal{Y}$. Recall that two sets $\mathcal{X}$ and $\mathcal{Y}$ are *equivalent* if and only if there exists a bijective (one-to-one and onto) map $f : \mathcal{X} \to \mathcal{Y}$. To address this problem, the concept of *isometry* for a set of points is needed.[8,12] In particular, an isometry is a diffeomorphism defined on a manifold such that it admits a local translation and rotation. The next definition extends the notion of isometry to data sets.

DEFINITION 2.1.[8] *Let $\mathcal{X} = \{x_1, x_2, \ldots, x_N\} \subset \mathbb{R}^D$ and $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^d$ be equivalent. Then $\mathcal{X}$ and $\mathcal{Y}$ are* $k$-locally isometric *if there exists a continuous map $T : \mathbb{R}^D \to \mathbb{R}^d$ such that if $T(x_i) = y_i$, then $T(N_{x_i}(k)) = N_{y_i}(k)$, $i = 1, \ldots, N$, where $N_{x_i}(k)$ (resp., $N_{y_i}(k)$) is the set of $k$-nearest neighbors of $x_i \in \mathcal{X}$ (resp., $y_i \in \mathcal{Y}$).*

Before stating the MVU method, we introduce the following maximization problem.

**Maximum Variance Unfolding Problem**. Given a set of input data points $\mathcal{X} = \{x_1, x_2, \ldots, x_N\} \subset \mathbb{R}^D$ find the set of output data points $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^d$, where $d \leq D$, such that the sum of pairwise square distances between the outputs in $\mathcal{Y}$ given by

$$\Phi = \frac{1}{2N} \sum_{i=1}^{N} \sum_{j=1}^{N} \|y_i - y_j\|^2, \tag{1}$$

is maximized, and $\mathcal{X}$ and $\mathcal{Y}$ are $k$-locally isometric for some $k \in \mathbb{Z}_+$.

Without loss of generality, we assume that $\sum_{i=1}^{N} x_i = 0$. Moreover, we require $\sum_{n=1}^{N} y_n = 0$ to remove the translational degree of freedom in the output points contained in $\mathcal{Y}$. Note that a data set (e.g., $\mathcal{X}$) can be represented by a weighted graph $\mathfrak{G}$,[13] where each node represents a point and the $k$-nearest points are connected by edges, where $k$ is a given parameter. The weights of $\mathfrak{G}$ represent the distance between the nodes. In addition, we assume that the graph $\mathfrak{G}$ is connected.[13] In the case of a disconnected graph, each connected component should be analyzed separately. The $k$-local isometry condition in the Maximum Variance Unfolding Problem

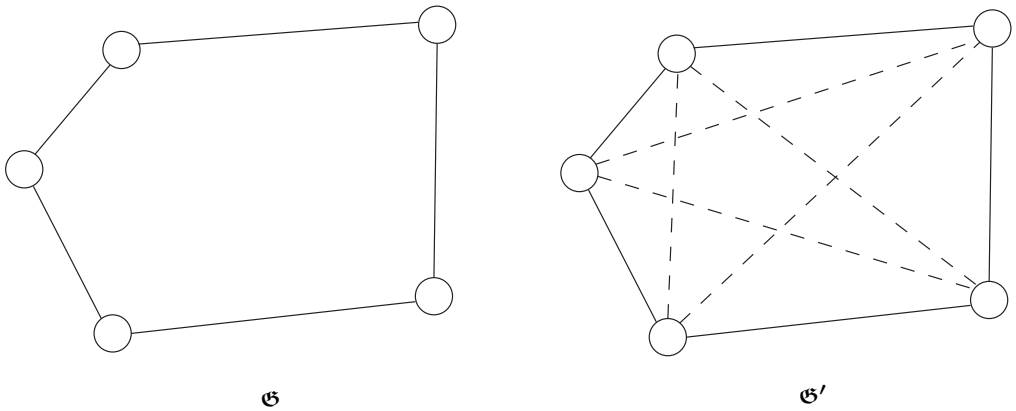


Figure 1. Original and modified graphs for $k = 2$.

requires that the distances and the angles between the $k$-nearest neighbors are preserved. This constraint is equivalent to preserving the distances between neighboring points in a modified graph $\mathfrak{G}'$, where, for each node, all the neighboring nodes of $\mathfrak{G}'$ are connected by an edge. More precisely, each node of $\mathfrak{G}'$ and the $k$-neighboring nodes of $\mathfrak{G}'$ form a *clique* of size $k+1$ (see Figure 1).[13]

The next theorem gives the solution to Maximum Variance Unfolding Problem for the case where $d = D$.

THEOREM 2.2.[8] *Consider the Maximum Variance Unfolding Problem with $d = D$. The output data points in $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^D$ are given by the solution to the optimization problem*

$$\max_{y_1, y_2, \ldots, y_N \in \mathbb{R}^D} \Phi, \tag{2}$$

*subject to*

$$\sum_{i=1}^{N} y_i = 0, \tag{3}$$

$$\|y_i - y_j\|^2 = D_{ij}, \quad \text{if } \eta_{(i,j)} = 1, \quad i, j = 1, \ldots, N, \tag{4}$$

*where $\Phi$ is defined in (1), $\eta = [\eta_{(i,j)}] \in \mathbb{R}^{N \times N}$ is the adjacency matrix of the modified graph $\mathfrak{G}'$, and*

$$D_{ij} = \|x_i - x_j\|^2, \quad i, j = 1, \ldots, N, \quad x_i, x_j \in \mathcal{X}. \tag{5}$$

The optimization problem (2)–(4) is *not* convex. The following convex optimization problem, however, is equivalent to the optimization problem given in Theorem 2.2. Moreover, the following result also addresses the case where $d \leq D$.

THEOREM 2.3.[8] *Consider the Maximum Variance Unfolding Problem with $d = D$ and let $\mathfrak{G}$ and $\mathfrak{G}'$ denote the weighted graph and modified graph corresponding to the data set $\mathcal{X}$, respectively. The output data points in $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^D$ are given by*

$$y_{ji} = \sqrt{\lambda_j V_{ji}}, \quad j = 1, \ldots, N, \quad i = 1, \ldots, D, \tag{6}$$

*where $V_{ji}$, $j = 1, \ldots, N$, $i = 1, \ldots, D$, is the $i$th component of the $j$th eigenvector of $K^*$ given by $V_j = [V_{j1}, V_{j2}, \ldots, V_{jD}]^{\mathrm{T}}$, $\lambda_j$ is the associated eigenvalue, $y_{ji}$, $j = 1, \ldots, N$, $i = 1, \ldots, D$, is the $i$th component of the vector $y_j = [y_{j1}, y_{j2}, \ldots, y_{jD}]^{\mathrm{T}}$, and $K^*$ is the optimal solution to the optimization problem*

$$\max_{K \in \mathbb{K}} \operatorname{tr}(K), \tag{7}$$

*subject to*

$$K \geq 0, \tag{8}$$

$$\sum_{i=1}^{N}\sum_{j=1}^{N} K_{(i,j)} = 0, \tag{9}$$

$$K_{(i,i)} - 2K_{(i,j)} + K_{(j,j)} = D_{ij}, \quad \text{if } \eta_{(i,j)} = 1, \quad i,j = 1,\ldots,N, \tag{10}$$

*where* $\mathbb{K} \subset \mathbb{R}^{N\times N}$ *denotes the cone of nonnegative definite matrices,* $\eta = [\eta_{(i,j)}] \in \mathbb{R}^{N\times N}$*, and* $D_{ij}$ *is defined as in (5). Moreover, if* $K^*$ *has* $d < D$ *non-zero eigenvalues, then the set of reduced dimension output data points is given by* $\mathcal{Y} = \{y_1^{\text{red}}, y_2^{\text{red}}, \ldots, y_N^{\text{red}}\} \subset \mathbb{R}^d$*, where* $y_i^{\text{red}}$*,* $i = 1,\ldots,N$*, is found by removing the zero elements of* $y_i$*.*

REMARK 2.4. *When the data is noisy, all the eigenvalues of* $K$ *are typically non-zero, and hence, one has to choose the dominant eigenvalues of* $K^*$*.*[10,11] *If the eigenvalues of* $K$ *are sorted in descending order, then the first* $d$ *components of* $y_i$*,* $i = 1,\ldots,N$*, form a* $d$*-dimensional data set that is* approximately $k$*-locally isometric to* $\mathcal{X} = \{x_1, x_2, \ldots, x_N\} \subset \mathbb{R}^D$*.*[8]

## 3. DATA SEGMENTATION AND SUBSPACE IDENTIFICATION

In this section, we address the problem of data segmentation and subspace identification for a given set of data points. First, we define the multiple subspace segmentation problem.

**Data Segmentation and Subspace Identification Problem**.[10,11] Given the set $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^D$, where $y_i$, $i = 1,\ldots,N$, are drawn from a set of distinct subspaces $\mathcal{S}_j$, $j = 1,\ldots,n$, of unknown number and dimension, find *i*) the number of subspaces $n$, *ii*) their dimensions $\dim(\mathcal{S}_j)$, and *iii*) the basis for each subspace. Furthermore, associate each point to the subspace it belongs to.

GPCA uses algebro-geometric concepts to address this problem. First, we present the GPCA algorithm followed by a more robust version of GPCA to deal with noisy data. For a detailed discussion, see Refs. 10 and 11.

### 3.1 Generalized Principal Component Analysis

In this subsection, we present the GPCA algorithm, where we assume that the data points are noise-free. The GPCA algorithm consists of three main steps; namely, polynomial fitting, polynomial differentiation, and polynomial division. The following definitions are needed.

DEFINITION 3.1.[10,14–16] *Let* $\mathcal{R}$ *be a commutative ring and let* $\mathcal{I}$ *be an additive subgroup of* $\mathcal{R}$*.* $\mathcal{I}$ *is called an* ideal *if* $r \in \mathcal{R}$ *and* $s \in \mathcal{I}$*, then* $rs \in \mathcal{I}$*. Furthermore, an ideal is said to be* generated *by a set* $\mathcal{S}$ *if, for all* $t \in \mathcal{I}$*,* $t = \sum_{i=1}^{n} r_i s_i$*,* $r_i \in \mathcal{R}$*,* $s_i \in \mathcal{S}$*,* $i = 1,\ldots,n$*, for some* $n \in \mathbb{Z}_+$*. Let* $\mathbb{F}[x]$ *be the set of polynomials of* $D$ *variables, where* $x = [x_1, x_2, \ldots, x_D]^{\text{T}}$*,* $x_i \in \mathbb{F}$*,* $i = 1,\ldots,D$*, and* $\mathbb{F}$ *is a field. Then* $\mathbb{F}[x]$*, with polynomial addition and multiplication, forms a commutative ring. A product of* $n$ *variables* $x_1, x_2, \ldots, x_n$ *is called a* monomial of degree $n$ *(counting multiplicity). The number of distinct monomials of degree* $n$ *is given by*

$$M_n(D) \triangleq C(D+n-1, n), \tag{11}$$

*where* $C(p,q)$ *denotes the combination of* $p$ *objects taken* $q$ *at a time. A polynomial with all of its terms being the same degree is called a* homogenous polynomial*. An ideal that is generated by homogenous polynomials is called a* homogenous ideal*. Finally, the* Veronese Map of degree $n$ *is a mapping* $\nu_n : \mathbb{F}^D \to \mathbb{F}^{M_n(D)}$ *that assigns to the variables* $x_1, x_2, \ldots, x_D$ *all the possible monomials of degree* $n$*; namely,*

$$\nu_n([x_1, x_2, \ldots, x_D]^{\text{T}}) = [u_1, u_2, \ldots, u_{M_n(D)}]^{\text{T}}, \tag{12}$$

*where* $u_i = x_1^{n_{i1}} x_2^{n_{i2}} \cdots x_D^{n_{iD}}$*,* $i = 1,\ldots,M_n(D)$*, and where* $n_{i1} + n_{i2} + \cdots + n_{iD} = n$*,* $n_{ij} \in \mathbb{Z}_+$*,* $j = 1,\ldots,D$*, and* $n_{i1},\ldots,n_{iD}$ *are in lexicographic order.*

Let $\mathcal{A} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_n\}$, $\mathcal{Z}_\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \cdots \cup \mathcal{S}_n$, where $\mathcal{S}_j \subset \mathbb{R}^D$, $j = 1, \ldots, n$, is a linear subspace, $\dim(\mathcal{S}_j) = d_j$, and $n \in \mathbb{Z}_+$. $\mathcal{A}$ is referred to as a *subspace arrangement*. In addition, let $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^D$ be a set of a sufficiently large number of points sampled from $\mathcal{Z}_\mathcal{A}$. In this paper, we assume that the number of subspaces $n$ is known. However, the GPCA algorithm, in its most general form, gives the solution for the case where $n$ is unknown.[10,11] In order to algebraically represent $\mathcal{Z}_\mathcal{A}$, we need to find the *vanishing ideal* of $\mathcal{Z}_\mathcal{A}$, denoted by $\mathcal{I}(\mathcal{Z}_\mathcal{A})$. The vanishing ideal of $\mathcal{Z}_\mathcal{A}$ is the set of polynomials which vanish on $\mathcal{Z}_\mathcal{A}$. It can be shown that the homogenous component of $\mathcal{I}(\mathcal{Z}_\mathcal{A})$, denoted by $\mathcal{I}_n$, uniquely determines $\mathcal{I}(\mathcal{Z}_\mathcal{A})$. Hence, to find the vanishing ideal $\mathcal{I}(\mathcal{Z}_\mathcal{A})$ it suffices to determine the homogenous component $\mathcal{I}_n$.

If $p_n(x)$ is a polynomial in $\mathcal{I}_n$, then $p_n(x) = c_n^{\mathrm{T}} \nu_n(x)$, where $c_n \in \mathbb{R}^{M_n(D)}$, $\nu_n(x) : \mathbb{F}^D \to \mathbb{F}^{M_n(D)}$ is the veronese map given by (12), $x = [x_1, x_2, \ldots, x_D]^{\mathrm{T}}$ for some $D \in \mathbb{Z}_+$, and $M_n(D)$ is given by (11). Therefore, each point $y_i$, $i = 1, \ldots, N$, satisfies $p_n(x) = 0$, and hence, $V_n(D)c_n = 0$, where

$$V_n(D) \triangleq \begin{bmatrix} \nu_n^{\mathrm{T}}(y_1) \\ \nu_n^{\mathrm{T}}(y_2) \\ \vdots \\ \nu_n^{\mathrm{T}}(y_N) \end{bmatrix} \tag{13}$$

is called the *embedded data matrix*. A one-to-one correspondence between the null space of $V_n(D)$ and the polynomials in $\mathcal{I}_n$ exists if

$$\dim(\mathcal{N}(V_n(D))) = \dim(\mathcal{I}_n) = h_\mathcal{I}(n), \tag{14}$$

or, equivalently,

$$\operatorname{rank} V_n(D) = M_n(D) - h_\mathcal{I}(n), \tag{15}$$

where the *Hilbert function* $h_\mathcal{I}(n)$ is the number of linearly independent polynomials of degree $n$ that vanish on $\mathcal{Z}_\mathcal{A}$. The singular vectors of $V_n(D)$ denoted by $c_{ni} \in \mathbb{R}^{M_n(D)}$, $i = 1, \ldots, h_\mathcal{I}(n)$, corresponding to the zero singular values of $V_n(D)$ can be used to compute a basis for $\mathcal{I}_n$, namely

$$\mathcal{I}_n = \operatorname{span}\{p_{ni}(x) = c_{ni}\nu_n(x),\ i = 1, \ldots, h_\mathcal{I}(n)\}.$$

In the case where the data set $\mathcal{Y}$ is corrupted by noise, the singular vectors corresponding to the $h_\mathcal{I}(n)$ smallest singular values of $V_n(D)$ are used to compute the basis for $\mathcal{I}_n$.

The following theorem shows how polynomial differentiation can be used to find the dimensions and bases of each subspace $\mathcal{S}_j$, $j = 1, \ldots, n$.

THEOREM 3.2.[10,11] *Let $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^D$ be a set of points sampled from $\mathcal{Z}_\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \cdots \cup \mathcal{S}_n$, where, for $j = 1, \ldots, n$, $\mathcal{S}_j$ is a subspace of unknown dimension $d_j$. Furthermore, assume that for every subspace $\mathcal{S}_j$, $j = 1, \ldots, n$, there exists $w_j \in \mathcal{S}_j$ such that $w_j \notin \mathcal{S}_i$, $i \neq j$, $i = 1, \ldots, n$, and condition (14) holds. Then*

$$\mathcal{S}_j^\perp = \operatorname{span}\left\{ \frac{\partial}{\partial x} c_n^{\mathrm{T}} \nu_n(x)|_{x=w_j} : c_n \in \mathcal{N}\left(V_n(D)\right) \right\}, \quad j = 1, \ldots, n, \tag{16}$$

*where $V_n(D)$ is the embedded data matrix of $\mathcal{Y}$ given by (13). Furthermore, $d_j = D - \operatorname{rank} \nabla P_n(w_j)$, $j = 1, \ldots, n$, where $P_n(x) = [p_{n1}(x), p_{n2}(x), \ldots, p_{nh_\mathcal{I}(n)}(x)]^{\mathrm{T}} \in \mathbb{R}^{1 \times h_\mathcal{I}(n)}$ is a row vector of independent polynomials in $\mathcal{I}_n$, composed of the singular vectors corresponding to the zero singular values of $V_n(D)$, and $\nabla P_n = [\nabla^{\mathrm{T}} p_{n1}(x), \nabla^{\mathrm{T}} p_{n2}(x), \ldots, \nabla^{\mathrm{T}} p_{nh_\mathcal{I}(n)}(x)] \in \mathbb{R}^{D \times h_\mathcal{I}(n)}$.*

To select a point $w_j$, $j = 1, \ldots, n$, for each subspace such that $w_j \in \mathcal{S}_j$, $w_j \notin \mathcal{S}_i$, $i \neq j$, $i = 1, \ldots, n$, without loss of generality, let $j = n$. One can show that the first point $w_n$, where $w_n \in \mathcal{S}_n$ and $w_n \notin \mathcal{S}_i$, $i = 1, \ldots, n-1$, is given by[10,11]

$$w_n = \operatorname{argmin}_{y \in \mathcal{Y}:\, \nabla P_n(y) \neq 0} P_n(y)(\nabla^{\mathrm{T}} P_n(y) \nabla P_n(y))^\dagger P_n^{\mathrm{T}}(y). \tag{17}$$

Furthermore, a basis for $\mathcal{S}_n$ can be found by applying PCA to $\nabla P_n(w_n)$. To find the rest of the points $w_i \in \mathcal{S}_i$, $i = 1, \ldots, n-1$, we can use polynomial division as outlined in the next theorem.

THEOREM 3.3.[10,11] *Let $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^D$ be a set of points sampled from $\mathcal{Z}_\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \cdots \cup \mathcal{S}_n$, where, for $j = 1, \ldots, n$, $\mathcal{S}_j$ is a subspace of unknown dimension $d_j$. Assume (14) holds, $\mathcal{S}_n^\perp$ is known, and a point $w_n \in \mathcal{S}_n$ is given. Then, the set $\bigcup_{j=1}^{n-1} \mathcal{S}_j$ is characterized by the set of homogenous polynomials given by*

$$\left\{ c_{n-1}^{\mathrm{T}} \nu_{n-1}(x) : V_n(D) R_n(b_n) c_{n-1} = 0,\, b_n \in \mathcal{S}_n^\perp,\, c_{n-1} \in \mathbb{R}^{M_{n-1}(D)} \right\},$$

*where $R_n(b_n) \in \mathbb{R}^{M_n(D) \times M_{n-1}(D)}$ is the matrix of coefficients of $c_{n-1}$ when $(b_n^{\mathrm{T}} x)(c_{n-1}^{\mathrm{T}} \nu_{n-1}(x)) \equiv c_n^{\mathrm{T}} \nu_n(x)$ is rearranged to have the form $R_n(b_n) c_{n-1} = c_n$.*

Once the homogenous polynomials $\{c_{n-1}^{\mathrm{T}} \nu_{n-1}(x)\}$ given by Theorem 3.3 are obtained, an identical procedure can be repeated to find $w_{n-1}$ and the homogenous polynomials characterizing $\bigcup_{j=1}^{n-2} \mathcal{S}_j$.

### 3.2 Subspace Estimation Using a Voting Scheme

The GPCA framework given in Subsection 3.1 works well in the absence of noise. In practice, however, noise is always present and efficient statistical methods need to be incorporated with the GPCA. In this subsection, we present one such statistical method where a voting scheme is combined with the GPCA. Here, we assume that the number of the subspaces and their dimensions are known. For details, see Refs. 10 and 11.

Let $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^D$ be the set of data points sampled from the set $\mathcal{Z}_\mathcal{A} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \cdots \cup \mathcal{S}_n$, where, for $j = 1, \ldots, n$, $\mathcal{S}_j$ is a subspace of dimension $d_j$ and co-dimension $c_j = D - d_j$. As noted in Subsection 3.1, the homogenous component of degree $n$ of the vanishing ideal $\mathcal{I}(\mathcal{Z}_\mathcal{A})$, denoted by $\mathcal{I}_n$, uniquely defines $\mathcal{I}(\mathcal{Z}_\mathcal{A})$ and $\dim(\mathcal{I}_n) = h_\mathcal{I}(n)$, where $h_\mathcal{I}(n)$ is the Hilbert function. Let $P = \{p_1(x), p_2(x), \ldots, p_{h_\mathcal{I}(n)}(x)\}$ be the set of polynomials forming a basis for $\mathcal{I}_n$. This set is obtained by selecting the $h_\mathcal{I}(n)$ smallest singular values of $V_n(D)$, where $V_n(D)$ is the embedded data matrix given by (13). Let $y_1 \in \mathcal{Y}$ and define $\nabla P_\mathrm{B}(y_1) \triangleq \left[\nabla^{\mathrm{T}} p_1(y_1), \nabla^{\mathrm{T}} p_2(y_1), \ldots, \nabla^{\mathrm{T}} p_{h_\mathcal{I}(n)}(y_1)\right]$. Note that in the noise-free case, if $y_1 \in \mathcal{S}_j$, then $\operatorname{rank} \nabla P_\mathrm{B}(y_1) = c_j$.

In the case where the data is corrupted by noise, a more efficient method for computing the basis is desired. Suppose the co-dimension of the subspaces $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_n$ take $q$ distinct values $c_1', c_2', \ldots, c_q'$, respectively. Given the fact that the membership of $y_1$ to one of the subspaces $\mathcal{S}_j$, $j = 1, \ldots, n$, is unknown, a set of basis candidates for the orthogonal complement of subspaces of all possible dimensions $c_i'$, $i = 1, \ldots, q$, is calculated by choosing the $c_i'$ principal components of $\nabla P_\mathrm{B}(y_1)$. This results in $q$ matrices $B_i \in \mathbb{R}^{D \times c_i'}$, $i = 1, \ldots, q$, each of which is a basis candidate for $\mathcal{S}_i^\perp$, $i = 1, \ldots, n$.

The main idea of the voting scheme is to count the number of repetitions of each basis candidate for all points in the data set $\mathcal{Y} = \{y_1, \ldots, y_N\}$. The $n$ basis candidates with the most votes are chosen to be the basis for $\mathcal{S}_i^\perp$, $i = 1, \ldots, n$, and each point is assigned to its closest subspace. In our criterion for counting the repetition of the basis candidates, two basis candidates are considered to be the same if the angle between the subspaces spanned by them is less than $\tau$, where $\tau > 0$ is a given tolerance parameter.

## 4. UNSUPERVISED LEARNING OF FACIAL EXPRESSIONS

The MVU and GPCA methods presented in Sections 2 and 3 can be used to recognize facial expressions from a given set of images within an unsupervised learning framework. Specifically, given a set of images of a person with two different facial expressions (e.g., neutral and happy), we show that the two facial expressions reside on two distinct subspaces if the manifold is unfolded. In particular, semi-definite embedding is used to reduce the dimensionality and unfold the manifold of facial images. Next, generalized principal component analysis is used to fit a series of subspaces to the data points and associate each data point to a subspace. Data points that belong to the same subspace are claimed to belong to the same facial expression. The algorithm is summarized in Table 1.

In our experiment, 30 photographs were taken for each human subject, where the subject starts by a neutral expression, transitions to a happy expression, and goes back to a neutral expression with each part containing

Table 1. Facial Expression Recognition Algorithm

**Step 1.** Preprocess of the grayscale image data $I_1, \ldots, I_N$.
- **a.** Compute $x_j \in \mathbb{R}^{D'}$ by column stacking the matrix of $I_j$, $j = 1, \ldots, N$.
- **b.** Set the number of neighbors $k$.
- **c.** Form the weighted graph $\mathfrak{G}$.
- **d.** Form the modified graph $\mathfrak{G}'$ and the adjacency matrix $\eta = [\eta_{(i,j)}]$.

**Step 2.** Manifold unfolding and dimension reduction.
- **a.** Set the dimension of the reduced space $D$.
- **b.** Find $K^*$, the maximizer of (7) subject to (8)–(10).
- **c.** Compute the eigenvectors and the associated eigenvalues of $K^*$; denote by $V_j = [V_{j1}, V_{j2}, \ldots, V_{jD'}]^{\mathrm{T}}$ and $\lambda_j$, $j = 1, \ldots, N$.
- **d.** Reorder $V_j$ and $\lambda_j$ such that $\lambda_j$, $j = 1, \ldots, N$ are in decreasing order.
- **e.** Compute $y_{ji} = \sqrt{\lambda_j V_{ji}}, \quad j = 1, \ldots, N, \quad i = 1, \ldots, D'$.
- **f.** Compute $y_j = [y_{j1}, y_{j2}, \ldots, y_{jD'}]^{\mathrm{T}}, \quad j = 1, \ldots, N$.
- **g.** Compute $y_j^{\mathrm{red}} = [y_{j1}, \ldots, y_{jD}]^{\mathrm{T}}, \quad j = 1, \ldots, N$.
- **h.** Compute $\mathcal{Y} = \{y_1^{\mathrm{red}}, \ldots, y_N^{\mathrm{red}}\}$.

**Step 3.** Subspace estimation using a voting scheme.
- **a.** Set the subspace angle tolerance parameter $\tau > 0$.
- **b.** For the subspaces $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_n$, compute the distinct value of their co-dimension $c_1', c_2', \ldots, c_q'$.
- **c.** Initialize the arrays $\mathcal{U}_1 = [\,], \ldots, \mathcal{U}_q = [\,]$ and the counters $\mathcal{C}_1 = [\,], \ldots, \mathcal{C}_q = [\,]$
- **d.** **for** $j = 1 : N$
- **e.** **for** $i = 1 : q$
- **f.** Compute the $c_i'$ principal components of $\nabla P_{\mathrm{B}}(y_j^{\mathrm{red}})$.
- **g.** Form the orthogonal matrix $B_i \in \mathbb{R}^{D \times c_i'}$ using outputs of **Step 3f.**
- **h.** **if** there exists $k$ such that the angle $\angle_{\mathrm{subspace}}(B_i, \mathcal{U}_i(k)) < \tau$, **then**
- **i.** $\mathcal{C}_i(k) \leftarrow \mathcal{C}_i(k) + 1$.
- **j.** **else**
- **k.** $\mathcal{U}_i \leftarrow [\mathcal{U}_i, B_i]$.
- **l.** $\mathcal{C}_i \leftarrow [\mathcal{C}_i, 1]$.
- **m.** **end for**
- **n.** **end for**
- **o.** Select basis candidates from $\mathcal{U}_1, \ldots, \mathcal{U}_q$ corresponding to the $n$ highest values of $\mathcal{C}_1, \ldots, \mathcal{C}_q$. Denote basis by $B_1, \ldots, B_n$.
- **p.** Use $B_1, \ldots, B_n$ (basis for $\mathcal{S}_1^{\perp}, \ldots, \mathcal{S}_n^{\perp}$) to find the $B_1', \ldots, B_n'$ (basis for $\mathcal{S}_1, \ldots, \mathcal{S}_n$).
- **q.** Use results in **Step 3p** to assign each $y_1^{\mathrm{red}}, \ldots, y_N^{\mathrm{red}}$, to the closest subspace.

10 photographs. An example set of images is given in Figure 2. These images were taken in a sequence, each $200 \times 240$ pixels, and in total there were 4 subjects.

Each image can be considered as a vector of dimension 48000 by column stacking the grey scale image intensity matrix. In this case, each image is a point in a 48000-dimension space. In order to segment the images, the dimension is reduced to $D = 5$ using the MVU algorithm. Then, the resulting points in the $D = 5$-dimensional ambient space are used to identify 2 subspaces of dimension $d = 1, 2, 3, 4$, where in the GPCA voting algorithm two subspaces are considered to be the same if the angle between the two subspaces is less than $\tau = 0.4$.[17] The segmentation error for each case is given in Table 2, where it is noted that the best results correspond to $d = 3$ and 4. In order to visualize the subspace identification, the segmentation for the case $D = 2$ and $d = 1$ is given in Figure 3. Although these parameters result in a poor segmentation performance, it graphically conveys the main idea of the algorithm.

Figure 2. A sequence of pictures, where the subject starts with a neutral expression, smiles, and resumes to a neutral expression.

Table 2. Segmentation Results for $D = 5$

| Subject | Number of Images | Segmentation Error | | | |
|---|---|---|---|---|---|
| | | $d = 1$ | $d = 2$ | $d = 3$ | $d = 4$ |
| 1 | 29 | 3 | 2 | 2 | 3 |
| 2 | 31 | 13 | 13 | 3 | 7 |
| 3 | 31 | 6 | 15 | 2 | 4 |
| 4 | 32 | 13 | 15 | 1 | 1 |

## 5. CONCLUSION

In this paper, we considered facial expression recognition within an unsupervised learning framework. Specifically, given a data set composed of a number of facial images of the same subject with different facial expressions, the algorithm introduced in this paper segments the data set into groups corresponding to different facial expressions. Each facial image can be regarded as a point in a high-dimensional space, and the collection of images of the same subject resides on a manifold within this space. Our results show that different facial expressions reside on distinct subspaces if the manifold is unfolded. In particular, semi-definite embedding was used to reduce the dimensionality and unfold the manifold of facial images. Generalized principal component analysis was used to fit a series of subspaces to the data points and associate each data point to a subspace. Data points that belong to the same subspace were shown to belong to the same facial expression. In future research we will extend the results to an unknown number of different facial expressions, and apply the framework to the problem of facial expression recognition for video imaging.

**Acknowledgements.** This research was supported in part by the US Army Medical Research and Material

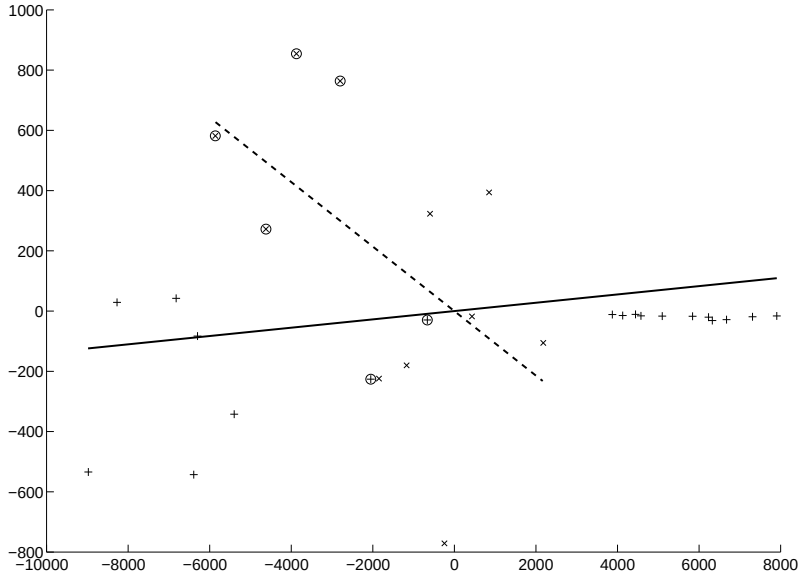


Figure 3. Facial expression segmentation with $D = 2$ and $d = 1$. The categorization error is 6/30. The solid and dashed lines are the subspaces corresponding to the neutral and happy expressions, respectively. The points associated with the solid line and the dashed line are depicted by "+" and "×", respectively. The points depicted by "○" are points associated with the wrong expression. Note that although these parameters result in a poor segmentation performance, it graphically conveys the main idea of the algorithm.

Command under grant 08108002 and by a grant from NIH (NAC P41 RR-13218) through Brigham and Women's Hospital. This work is part of the National Alliance for Medical Image Computing (NAMIC), funded by the National Institutes of Health through the NIH Roadmap for Medical Research, Grant U54 EB005149.

## REFERENCES

[1] Ekman, P., [*Emotion in the Human Face*], Cambridge University Press, New York, NY (1982).

[2] Pantic, M. and Rothkrantz, L. J. M., "Toward an affect-sensitive multimodal humancomputer interaction," *Proc. IEEE* **91**, 1370–1390 (2003).

[3] Brahnam, S., Nanni, L., and Sexton, R., "Introduction to neonatal facial pain detection using common and advanced face classification techniques," *Stud. Comput. Intel.* **48**, 225–253 (2007).

[4] Haddad, W. M. and Bailey, J. M., "Closed-loop control for intensive care unit sedation," *Best Pract. Res. Clin. Anaesth.* **23**, 95–114 (2009).

[5] Gholami, B., Haddad, W. M., and Tannenbaum, A. R., "Agitation and pain assessment using digital imaging," *Proc. IEEE Eng. Med. Biolog. Conf.*, 2176–2179 (2009).

[6] Gholami, B., Haddad, W. M., and Tannenbaum, A. R., "Relevance vector machine learning for neonate pain intensity assessment using digital imaging," *IEEE Trans. Biomed. Eng.* (submitted).

[7] Chang, Y., Hu, C., Feris, R., and Turk, M., "Manifold based analysis of facial expression," *Imag. Vis. Comput.* **24**, 605–614 (2006).

[8] Weinberger, K. Q. and Saul, L. K., "Unsupervised learning of image manifolds by semi-definite programing," *Int. J. Comp. Vis.* **70**, 77–90 (2006).

[9] Jolliffe, I. T., [*Principal Component Analysis*], Springer (2002).

[10] Vidal, R., Ma, Y., and Sastry, S., [*Generalized Principal Component Analysis*], Springer-Verlag (2005).

[11] Vidal, R., Ma, Y., and Sastry, S., "Generalized principal component analysis (GPCA)," *IEEE Trans. Patt. Anal. Mach. Intell.* **27**, 1945–1959 (2005).

[12] Tenenbaum, J. B., de Silva, V., and Langford, J. C., "A global geometric framework for nonlinear dimensionality reduction," *Science* **290**, 2319–2323 (2000).
[13] Thulasiraman, K. and Swamy, M. N. S., [*Graphs: Theory and Algorithms*], Wiley-Interscience (1992).
[14] Eisenbud, D., [*Commutative Algebra*], Springer-Verlag (1995).
[15] Tannenbaum, A. R., [*Invariance and System Theory: Algebraic and Geometric Aspects*], Springer-Verlag (1981).
[16] Gallian, J. A., [*Contemporary Abstract Algebra*], D. C. Heath and Company (1990).
[17] Yang, A. Y., Rao, S., Wagner, A., Ma, Y., and Fossum, R. M., "Hilbert functions and applications to the estimation of subspace arrangements," *Int. Conf. Comp. Vis.*, 158–165 (2005).